COMMODORE USERS

Ahoy!

NOW 146 PAGES!

$2.50/ CAN. $3.00 OCT. 1985

...FIRST LOOK AT COMMODORE'S AMIGA!...

PROGRAM **CARTOON CHARACTERS** INTO YOUR C-64 GAMES

AMIGA
CHOICE OF THE NEXT GENERATION?
FIRST AVAILABLE PHOTOS

A HOWL! **ART GALLERY** OF HOMEGROWN HALLOWEEN GRAPHICS

DISK DRIVE UTILITIES!
- ERROR CHECKER
- 1541 CLEANING PROGRAM

SHADEYDUMP
CHIAROSCURO PRINTOUTS OF MULTICOLOR SCREENS
SHOTGUN
HIGH CALIBER MENU MAKER

GAMES!
- MYSTERY AT MYCROFT MEWS
- GRAVINAUTS
- SOLITAIRE

INFINITESIMAL INTRIGUE
SUB-MICROSCOPIC SIMULATIONS ON YOUR 64

BIT BY BIT PROGRAMMING
PERFORMING SINGLE BIT MANIPULATIONS ON BINARY NUMBERS

BEACH-HEAD II REVIEWED
PLUS ASSEMBLY LANGUAGE AND BEGINNER'S COLUMNS AND MORE...END



48355 K
10
0 71896 48355

Ahoy!

CONTENTS

DEPARTMENTS

FEATURES

PROGRAMS

Cover art by James Regan; art inside screens by James Regan, Tom Cushwa, Sandra Steele

Publisher
Michael Schneider

Editor
David Allikas

Managing Editor
Michael Davila

Senior Editor
Tim Moriarty

Technical Editors
David Barron
Bob Lloret

Consulting Editors
Ben Bova
Morton Kevelson
Dale Rupert

Art Director
JoAnn Case

Production Director
Melissa Held

Art Production
Christopher Carter
Mark Kammerer
Victoria Green

Bulletin Board SYSOP
B.W. Behling

Circulation Director
W. Charles Squires

Advertising Director
Lynne Dominick

Director of Promotion
Joyce K. Fuchs

Controller
Dan Tunick

Managing Director
Richard Stevens

Advertising Representative
JE Publishers' Representative
6855 Santa Monica Blvd., Suite 200
Los Angeles, CA 90038
_(213)467-2266_
Boston (617)437-7628
Dallas (214)660-2253
New York (212)724-7767
Chicago (312)445-2489
Denver (303)595-4331
San Francisco (415)864-3252

ISSUE NO. 22 OCTOBER 1985

Ahoy! (#8750-4383) is published monthly by Ion International Inc., 45 W. 34th St., Suite 407, New York, NY, 10001. Subscription rate: 12 issues for $19.95, 24 issues for $37.95 (Canada $26.95 and $49.95 respectively). Second class postage paid at New York, NY 10001 and additional mailing offices.

No responsibility can be accepted for unsolicited material. Postmaster, send address changes to _Ahoy!_, 45 W. 34th Street, Suite 407, New York, NY 10001. Direct all address changes or matters concerning your subscription to _Ahoy!_, P.O. Box #341, Mt. Morris, IL 61054. All editorial inquiries and software and hardware to be reviewed should be sent to _Ahoy!_, 45 W. 34th St., Suite 407, New York, NY 10001.

VIEW FROM THE BRIDGE

A competitor of ours recently ran an editorial page explaining why their magazine had shrunk to 96 pages. Not that any explana tion was really necessary--obviously, business had gone a little sour. Nor is any explanation really necessary for us this issue as we jump to 146 pages--business is going through the roof!

And we're not through yet. Premiering next issue will be an eight-page monthly addition that will make the best Commodore magazine on the market better still!

In order to provide our copious monthly collection of programs and articles geared to the more serious Commodore user, we've historically neglected some of the elementary aspects of Commodore home computing–among them, entertainment software. But now that we've grown in size we're going to correct that fault, with a monthly entertainment software section edited by *Ahoy!* game reviewer Arnie Katz, founder and former editor of *Electronic Games* magazine. Beginning next month, *Ahoy!* will feature more complete game coverage than any other Commodore magazine!

For this month, though, we'll have to content ourselves with leading the field in every *other* aspect of Commodore home computing, with articles and programs like the following:

Continuing his series on sprite programming, *Creating Your Own Games* columnist Orson Scott Card tells you how to create animated cartoon characters that are *Alive and in Living Color.* (Turn to page 18.)

Dale Rupert continues doing some of the most original work in computer magazines today with *Infinitesimal Intrigue,* computer simulations of sub-microscopic activity. (Turn to page 51.)

As the debate over copy protection continues, let's take a month off to learn what makes it tick with George Trepal's *Copy Protection: The Guarded Secrets.* Included is *Disk Error Checker,* a program that allows you to examine copy protection schemes. (Turn to page 59.)

It may sound like a cheap motel, but *ShadeyDump* is actually a boon to graphic artists, permitting them to dump multicolor screens in shades of black and white. (Turn to page 93.)

Mystery at Mycroft Mews is a murderously entertaining adventure game for the 64. (Turn to page 29.)

Shotgun takes its name from the speed with which programs can be loaded and run from the disk menu it generates. (Turn to page 97.)

The 1541 Disk Drive Cleaning Utility is a necessary complement to the many drive-cleaning disks on the market. (Turn to page 107.)

The leading playing card manufacturers begged us not to print it, but *Solitaire* appears in this issue anyway. It plays so much like the real game, you may never buy another deck of cards! (Turn to page 37.)

Gravinauts requires you to fight for survival in the hostile environment of the planet Arkion. (Turn to page 32.)

Maestro simplifies making and editing music on the C-64. (Turn to page 39.)

Mark Andrews continues to put the power of machine language under your control with this month's installment of Commodore Roots, dealing with performing single bit manipulations on binary numbers. (Turn to page 103.)

Our popular *Cadet's Column* continues this issue, with advice to beginning Commodore computerists on organizing their workspace while the organizing is good. (Turn to page 109.)

Also inside is a special *Scuttlebutt* report on Commodore's new Amiga computer; a modest first installment of *Tips Ahoy!,* our new programming hints column; *Reviews* of *Beach-Head II, Isepic, FAST-HACK'EM,* and other recent Commodore releases; the latest monthly batch of *Commodares;* and many other surprises.

Meanwhile, we dedicate this largest issue in *Ahoy!* history to you, the loyal readers who made our success happen. As long as you continue to support us, we'll continue to support you, with the best we can deliver, month after month. *–David Allikas*

NEW LOW PRICES FROM THE WORLD'S LARGEST SPORTS SOFTWARE COMPANY!

Whether you're a coach, player, computer bug, statistician, dabbler, doer or just a real sports fan, PDS has a package for you...at our lowest prices ever! Check our line-up below, and then order your first PDS Sports Package.



FOOTBALL STATISTICS. A comprehensive computer software system. You compare teams in over 60 offensive and defensive situations and watch the matchups on the screen. Every team is given a power rating. PDS also provides "year-to-date" data diskettes for a nominal charge. 39.95

FOOTBALL HANDICAPPING. (Includes FOOTBALL STATISTICS Package). Forecasts the outcome of football games based on massive amounts of statistical data, morning lines and point spreads. 49.95



BASKETBALL STATISTICS. The most comprehensive basketball program ever written. "What-if" your way into every imaginable comparison and formulation. 39.95

BASKETBALL HANDICAPPING. (Includes BASKETBALL STATISTICS). Gives you an inside look into the outcome of games by mathematically equating statistics, lines and point spreads. 49.95



BASEBALL STATISTICS. See how opposing baseball teams stack up against each other. See summarizations. Performance stats on every team in the major leagues. 39.95

BASEBALL HANDICAPPING. (Includes BASEBALL STATISTICS) Makes you better-prepared to predict the winner of any game. At least, mathematically. 49.95



THOROUGHBRED HANDICAPPING. This "world class" program gives you an edge in predicting winners. A proven system...by PDS and many horseracing enthusiasts. 129.00



HARNESS RACE HANDICAPPING. The fast, easy way to find out what races and what horses should be looked at...all summarized in a "power rating" format. 129.00



QUARTER HORSE HANDICAPPING. the computer system actually "rates" the horses in each race of what is the fastest growing segment in the sport of kings. 129.00



TRAINER STATISTICS. Analyzes all local racehorse trainers and gives you a rating for each one. 39.95



JOCKEY STATISTICS. A short time with this software package will show you why the top 15 jockeys at local tracks win over 90% of all races. 39.95

HARNESS DRIVER STATISTICS. A complement to the HARNESS HORSE HANDICAPPING System, this package tells you all you need to know about the man (or woman) *behind* the horse. 39.95



PDS SPORTS™

P.O. BOX E / TORRANCE, CA 90507 / (213) 516-6688

Please send me the following PDS SOFTWARE PACKAGE:

____________________...$________

____________________...$________

____________________...$________

Total

Add $6.00 for postage and handling.

Calif. residents add 6½% sales tax

I have an APPLE ( ) IBM ( ) TRS-80 ( ) COMMODORE-64 ( )

DISKETTE ( ) CASSETTE ( ); MODEL # ______

( ) Check enclosed

( ) Charge my credit card: A.E. ( ), VISA ( ), M.C. ( )

Card No.____________________Expires______

Signature____________________

(As it appears on credit card)

OR CHARGE BY PHONE...CALL (800) 222-2601 (In Calif: (213) 516-6688)

NAME____________________

ADDRESS____________________PHONE__________

CITY______________STATE______ZIP______

Dealer Inquiries Invited

Reader Service No. 168

PAPER AIRPLANE PROGRAM • C-128 80-COLUMN ADAPTER • COMPUSERVE CHANGES • GAMES FROM INFOCOM, MICROPROSE, SSI • HOME BUDGET PROGRAMS • $4.95 SOFTWARE • SUPER PASCAL • ENERGY ANALYZER • MIDI PRODUCTS • SPEECH SYNTHESIZER • BOOKS • AMIGA SOFTWARE

UNVEILING THE AMIGA

By Tom Benford / *Ahoy!* Business Press Correspondent

NEW YORK (July 24, 1985)–The long-awaited Commodore Amiga made its official debut last night. The coming-out party was very similar to a debutante's ball–the trappings and festivities of the occasion fully befitted this dazzling new machine.

The event was held in Manhattan's Lincoln Center. Several celebrities, including Andy Warhol and Debbie Harry, participated in showing off the machine's fantastic graphic capabilities. But the real star of the evening was the Amiga itself. The power, features, and performance of the machine are all truly stellar in scope. It was exciting to be there, to be among the first people to get a look at what personal computing will be like in the future.

The Amiga is obviously intended to do battle with the Macintosh and IBM-PC/AT in the personal computer market. Several speakers at the presentation mentioned superior features of the Amiga in direct comparison to these competitive machines. The Amiga's creators have endowed it with impressive armor that should be able to withstand repeated jousts with Big Blue and the Mac.

Unlike the PETs, VIC, C-64, Plus/4, C-16, "B" series, or even the C-128 computers, which utilize an 8-bit microprocessor, the Amiga's heart is a Motorola 68000 16/32-bit CPU–the same as the Macintosh. The PC uses an 8/16-bit chip, the Intel 8088; the AT is driven by an 8/24-bit CPU, the Intel 80286.

The Mac and IBMs rely heavily on the CPU to handle many mundane chores, such as graphics mapping, controlling the I/O ports, and managing keyboard input and functions; these necessary tasks all sap strength, speed, and power from the CPU. Since less of the CPU's power is available for actual computations, the microprocessor is never fully available to the user–he must share the CPU with the machine itself. The Amiga makes a radical departure from such "traditional" task-networking at this point; the prospect of having the 68000 totally available for use is positively awesome. The Amiga's heart is strong, fast (7.8 MHz), and accessible.

The soul, however, comes from three independent, dedicated VSLI chips. One handles the audio/sound capabilities, disk control, and other I/O functions; another chip handles the graphics screen mapping and sprites; the third custom chip takes care of the Amiga's dual animation capabilities. All the while the CPU is barely used, which leaves its power available for data handling and computing–this remarkable engineering feat is what sets the Amiga apart from the competition.

But that's only the tip of the iceberg. Though jaded from witnessing the introduction of several "revolutionary" computers first-hand, I was genuinely excited by the Amiga. And the more I saw, the more excited I grew. With the introduction of the Amiga, we're into a whole new ballgame of personal computing. "Wow!" was an utterance used very frequently by myself and others who attended the Amiga's premier. And the "wows" continued throughout the 2-hour presentation of the machine; the more I saw and heard, the better it got.

Out of the box the Amiga comes with 256K RAM and 192K ROM. Like the Mac, the Amiga's RAM can be "hardware" expanded to 512K (Tecmar already has a 1MB expansion module available); it also handles external hard-disk expansion up to 8MB. Wow!

The Amiga comes equipped with a built-in 3.5" disk drive which has a formatted capacity of 880K, more than double that of the Mac and PC. Additionally, RS232, parallel, and serial ports are all standard on the machine. Wow!

Three video output formats are supported by the Amiga: RGB, composite, and color TV; the high-resolution mode yields an incredible 640x 400 pixel display; any 16 of the Amiga's 4,096 colors may be displayed at one time. It's important to note that multiple windowing is also supported, and each window may display its own set of 16 colors; an impressive demonstration of this capability displayed all 4,096 colors on the screen simultaneously. Double wow!

In the music and sound depart-

ments, the Amiga is in a class by itself. A built-in four-voice synthesizer with stereophonic output capability only belies the aural goodies available within the machine. Add built-in voice synthesis with unlimited text-to-speech capabilities, MIDI compatibility, full polyphonic capabilities with superior synthesis features, and you have a machine which the competition can't even come close to. The Mac can support four voices by sapping the CPU heavily; the PC and AT offer only single-voice sound. None have speech synthesis capabilities. Wow!

An 89-key keyboard with numeric keypad is the standard input device on the Amiga; in contrast, the Mac sports only 58 keys while the PC and AT provide 82 and 84 keys respectively. The Amiga also handles input from a two-button mouse, joystick, light pens, and a host of external input devices including video digitizers, audio A/D (analog to digital) converters, MIDI-equipped instruments, and numerous other input/control devices either already available or nearing completion.

Microsoft developed the powerful dialect of BASIC supplied with the Amiga. All of the computer's abundant features and capabilities—sound, music, speech, graphics, animation, windowing, etc.—are directly accessible through BASIC programming. Additional programming languages for the Amiga include Pascal, C, Logo, Lisp, Assembler, and others. The machine is very flexible as to which programming language is used, since all languages—including BASIC—are disk-based and are booted into the system prior to use.

Another highly innovative feature of the Amiga is that it uses a "kick-start" disk to configure and control the machine. Unlike a boot disk which places the operating system directives into RAM memory locations, the "kick-start" disk addresses the three custom-designed VSLI chips solely, leaving the machine's entire 256K of RAM available for use. Wow, again!

Since the CPU's computing power is available to do useful work at all times, multitasking is a powerful reality on the Amiga. An impressive demonstration of multitasking was provided by having word processing, data sorting, animated graphics, business graphics, sound, and spreadsheet programs all running simultaneously within the Amiga's multiple windows. While we witnessed this very busy display, we were told that less than 9% of the 68000 CPU's power was being utilized—trying to imagine what you could do with the remaining 91%+ simply boggles the mind. No wow—utterly speechless this time!



The Amiga includes 256K of ROM, built-in 3.5" drive (format capacity of 880K), and parallel, serial, and RS232 ports. It has 4 voices, 4096 colors, and a 640X400 pixel display. But can it type? Yes—with an 89-key keyboard that includes a numeric keypad. Scheduled for September release, at a price of $1295. READER SERVICE NO. 192

With a suggested retail price of $1,295 and September availability, the Amiga has without a doubt broken new ground on the scene of personal computing. Optional equipment including an ultra-high-res monitor, hard-disks, tape back-ups, modems, laser videodisc controllers, video interfaces and digitizers, and external musical keyboards, promise to shatter the creative chains that have hindered computers and users in the past.

The Amiga can even be de-tuned to emulate an IBM. A demonstration of this was provided in grandiose style in keeping with the theme of the evening. A PC Emulator program was booted via the 3½" built-in drive and, lo and behold, the PC-DOS message visible on the screen indicated that the Amiga had just changed its normal way of doing things to conform to the IBM standard. Another disk was inserted into the drive, and the word "lotus" entered at the "A>" prompt. We waited in suspense for several seconds while the screens of the three large video monitors in the theatre remained blank. Then a comment from Bob Parasal, one of the Amiga's patriarchs: "Lotus 1-2-3 takes just as long to load on the Amiga when it is emulating a PC as it does on a PC!" The opening screen and spreadsheet video display of Lotus's best-selling program became visible on the giant screens. Parasal commented dryly, "You'll notice the display looks identical to the PC...same flavor...standard IBM vanilla," referring to the white characters on black background.

Software support was evidenced by such companies as Arktronics, Broderbund, Mindscape, Electronic Arts, EasyWare, Metacomco, Cherry Lane Technologies and others (see below). The features and creative

freedom the Amiga affords programmers are such to make it a hit among software developers, which will insure its success in the marketplace.

As far as pushing the state-of-the-art in personal computers goes, the Amiga is what the world—myself included—has been waiting for. Congratulations to everyone who had a hand in making the Commodore Amiga a reality, and good luck on your beautiful new baby.

AMIGA SOFTWARE

The New York Times reported that the Amiga's premier was attended by a meager contingent of committed developers. We can only assume that the reporter who covered the event for that esteemed publication was attending his first computer introduction. The amount of software and hardware displayed or announced at the Amiga's coming out party is staggering compared to that available for any new micro within memory—so much that we have room for only the briefest description of each.

Available upon Amiga introduction:

The Amiga Eye, a color video digitizer of monochrome or color images that lets the user vary hue, saturation, and luminance, as well as brightness over the computer's range of 4,000-plus colors. (A-Squared Systems Group, 7200 Sayre Drive, Oakland, CA 94611.)

Textcraft, a word processor with four type styles, three fonts, and six templates of pre-formatted documents. (The Arktronics Corporation, 520 East Liberty St., Ann Arbor, MI 48104.)

Graphicraft, a paint and art program driving 32 high resolution colors that lets users airbrush with variable density and diameter, spread color, repeat objects, and cycle colors; *Moviecraft*, an animated drawing program that uses a "tweening" technique to produce animated sequence without having to read from a disk; and *Presentationcraft*, a business graphics program that lets you produce 3D objects, and explode and expand bar and pie charts. (Island Graphics Corporation, One Harbor Dr., Sausalito, CA 94965.)

Lattice C Compiler, allowing programs designed for other PC operating systems to work on the Amiga; *Lattice C Cross Compiler/IBM MS-DOS*, allowing Amiga programs to run on the IBM PC; *Lattice C Cross-Compiler/UNIX* (Amiga programs on UNIX), *Lattice C Cross-Compiler/VAX* (Amiga programs on VAX); *LMK*, a development tool similar to UNIX-Mate; *LSE*, a multi-language screen editor; and *TMN*, providing text management utilities for searching simple or complicated files. (Lattice, Inc., P.O. Box 3072, Glen Ellyn, IL 60138.)

TLC-Logo, for programming in LISP and AI (artificial intelligence). (The Lisp Company, 430 Monterey Ave., #4, Los Gatos, CA 95030.)

Amiga Assembler/Linker, a Motorola standard macro-assembler with a corresponding linker; *Cambridge LISP 68000*, a language for use in artificial intelligence; *Metacomco BASIC*, a BASIC interpreter; *Amiga DOS*, a multitasking operating system; and *MCC Pascal 6800*, a full ISO 7185 standard Pascal single pass compiler. (Metacomco, 201 Hoffman Ave., Monterey, CA 93904.)

ABasiC, a language exploiting most of the Amiga's advanced features, including multiple windows, animation, four-voice sound, and menus. (Microsoft Corporation, 10700 Northup Way, Bellevue, WA 98009.)

Deja Vu, a graphics text adventure set in 1940's Hollywood; and *Amiga Tutor*, a chapter-by-chapter look at the machine's major features. (Mindscape, Inc., 3444 Dundee Road, Northbrook, IL 60062.)

Enable/Write, an advanced word processor for professional users. (The Software Group, Northway Ten Executive Park, Ballston Lake, NY 12019.)

T-Card, an expansion card with up to 1MB of memory featuring a serial port, clock/calendar with battery backup, hard disk interface, parallel printer port, and additional power supply; *T-Disk*, a 20MB 3.5-inch hard disk drive; *T-Modem*, Hayes-compatible with 300, 1200, and 2400 baud rates; and *T-Tape*, a 20MB tape backup for a hard disk drive. (Tecmar, 6225 Cochran Rd., Solon, OH 44139.)

The following were scheduled for August release:

RAGS to RICHES Ledger, a double-entry ledger handling up to 5,000 accounts/10,000 transactions. (Chang Laboratories, Inc., 5300 Stevens Creek Blvd., San Jose, CA 95129.)

Harmony, utilizing artificial intelligence to create musical accompaniment that follows the musician's tempo. (Cherry Lane Technologies, 110 Midland Ave., Port Chester, NY 10573.)

Penmouse Input Device, emulating both a mouse and a graphics tablet. (Kurta Corporation, 4610 South 35th St., Phoenix, AZ 85040.)

Scheduled for September release:

RAGS to RICHES Payables and *RAGS to RICHES Receivables*, which can integrate with each other and with *Ledger*, described above. (Chang Laboratories, Inc.)

Dr. J & Larry Bird Go One-on-One and *Seven Cities of Gold*, en-

hanced version of the popular games. (Electronic Arts, 2755 Campus Dr., San Mateo, CA 94403.)

The following from the Infocom canon: *Zork I, II,* and *III, Enchanter, Sorcerer, Suspect, The Witness, Cutthroats, Deadline, Seastalker, Infidel, Planetfall, Suspended, Starcross, Invisi Clues,* and *The Hitchhiker's Guide to the Galaxy.* (Infocom, Inc., 125 CambridgePark Drive, Cambridge, MA 02140.)

CalCraft, a spreadsheet with pull-down menus and flexible formatting options. (Synapse Software, 5221 Central Ave., Richmond, CA 94804.)

RAGS to RICHES Sales, a register program for point-of-sale income accounting (integratable with other *RAGS to RICHES* programs described above). (Chang Laboratories, Inc.)

Texture, a modular music recording program. (Cherry Lane Technologies.)

Archon and *Skyfox,* two more game adaptations. (Electronic Arts.)

Musicraft, which turns the Amiga into a synthesizer and sequencer. (Everywhere, Inc., P.O. Box 3418, Northridge, CA 91323.)

Enable/Calc, a spreadsheet with Lotus-like macros and over 50 math functions. (The Software Group.)

Telecraft, a communications and terminal emulation package. (Software 66, 3119 East Des Moines, Mesa, AZ 85203.)

Scheduled for November release:

The Print Shop, presumably an enhanced version. (Broderbund Software, 17 Paul Dr., San Rafael, CA 94903.)

Deluxe Video Construction Set, for creating animated business graphics with sound effects; and *Marble Madness* and *Return to Atlantis.* (Electronic Arts.)

Enable/File, a database manager that can handle up to 1301 bytes of information with over 26 fields per record. (The Software Group.)

Scheduled for December release:

Enable/The Office Manager, consisting of integrated word processing, spreadsheet, database, telecommunications, and graphics programs. (The Software Group.)

Scheduled for the nebulous "4th quarter 1985":

The Helley Project: A Mission in Our Solar System, playable with joystick or mouse; and *Keyboard Cadet.* (Mindscape, Inc.)

Radar Raiders, a flight simulation game with 3D imaging, color graphics with shading, hidden surface removal and patterning in real time. (SubLOGIC Corporation, 201 West Springfield Ave., Suite 1001, Champaign, IL 61820.)

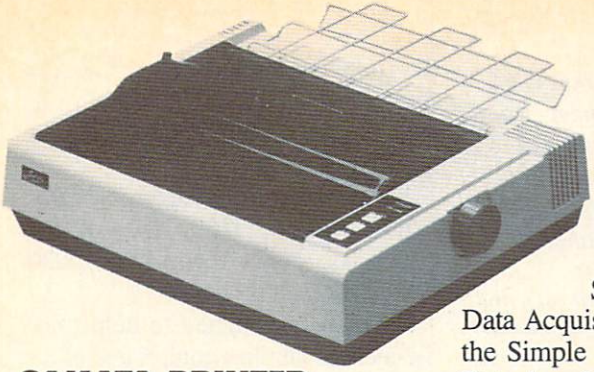
Scheduled for 1st quarter 1986:

Turbo Pascal, a single-pass, native code compiler. (Borland International, 4585 Scotts Valley Drive, Scotts Valley, CA 95066.)

Sargon III, chess simulation with nine levels of play. (Hayden Software Company, 600 Suffolk St., Loweil, MA 01854.)

And for January 1986:

Scorewriter, allowing musicians to score and print music. (Cherry Lane Technologies.)



SP-1200 Plus has German, Spanish, Italian, Swedish, Danish fonts. *READER SERVICE NO. 193*

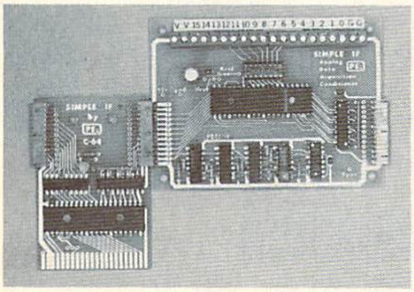
SAKATA PRINTER

Sakata's SP-1200 Plus printer offers 120 character per second speed, double-print and near letter quality modes, and the capacity to print eight international character fonts. Price will be $359.00.

Sakata U.S.A. Corporation, 651 Bonnie Lane, Elk Grove Village, IL 60007 (phone: 312-593-3211).

OUTSIDE WORLD INTERFACES

Two devices designed to link your Commodore computer with the outside world:



Proteus' Simple IF & conditioner. *READER SERVICE NO. 194*

With an X-10 Powerhouse in your C-64's user port, you can automatically control up to 72 lights, appliances, heating and air conditioning units, and other electrical devices. The unit contains its own microprocessor, and memory backed up by a battery that can sustain it without AC power for more than 100 hours. Price of interface, software disk, and connecting cable is $120.

X-10 (USA) Inc., 185A Legrand Ave., Northvale, NJ 07647 (phone: 201-784-9700).

Enabling VIC 20, C-64, and C-128 users to digitize 16 channels of analog signals (and thus use their computers to perform heating/cooling/solar control, voltage measurement, weather station monitoring, and other tasks), the Simple Interface Analog Data Acquisition System consists of the Simple IF card and the Analog Data Acquisition Conditioner. Price is under $100; demo disk is $9.95. Other plug-in conditioning boards are also available.

Proteus Electronics Inc., Spayde Road RD2, P.O. Box 693, Bellville, OH 44813 (phone: 419-886-2296).



X-10 Powerhouse handles 72 appliances. *READER SERVICE NO. 195*

C-128 80-COLUMN ADAPTER

Cardco's C-128/80 video adapter cable will enable the Commodore 128 to output 80-column monochrome displays to any standard composite video monitor. (The unaided C-128 will output to non-RGB monitors only in 40-column mode.) The 5' cable, which connects to the computer's RGB port and the monitor's video input jack, will sell for $9.95.

Also coming from Cardco: a 32K printer buffer that wil return the computer to the C-64, C-128, C-16, or Plus/4 user up to 15 times sooner than when you send data directly to the printer.

Cardco, Inc., 300 S. Topeka, Wichita, KS 67202 (Phone: 316-267-6525).

ENGINEERING PROGRAM

Intended to make life easier for engineers, technicians, and students, BAMTEK's *Professional Electronic Engineering Software* for the C-64 provides electronic circuit analysis, neg/pos feedback active filters, frequency response plots, and a polynomial root finder. Price is $39.95 plus $1.95 shipping (CA residents add 6% sales tax).

BAMTEK, 1541 N. China Lake Blvd., Suite 584, Ridgecrest, CA 93555 (phone: 619-446-5311/ext. 584).

$4.95 SOFTWARE

Hoping to singlehandedly drive software pricing down several strata, Green Valley Publishing is offering a number of disks for the C-64, each containing three programs, at $4.95-$9.95 each. Available at K-Mart (under the name Load 'N' Go!), Target Stores (as One-Step Software), and other chains, current titles include the following:

Games: *Championship Gambler, Bits, Pieces & Clues, Board Game Challengers, Star Trek Evolution, Maze Madness, Board Games I, Chess Champion, Adventure Master.*

Utilities: *Home Financial Organizer I, My C-64 Computer Tutor, Pro Financial Organizer II, Home Property Manager, Home Income Manager, Personal Spreadsheet, Personal File Keeper, Recipe Box, Word Writer.*

Green Valley Publishing, Inc., 7122 Shady Oak Road, Minneapolis, MN 55344 (phone: 612-829-0398).

BOOK RELEASES

Commodore 64 Assembly Language Arcade Game Programming includes advice on topics like the use of macros and subroutines, and entering graphics data by use of a graphics tablet. Two sample games are included. $14.45 from TAB Books Inc., Blue Ridge Summit, PA 17214 (phone: 717-794-2191).

The *Minute Manual for the Dot Matrix Printer* guides the non-programmer through buying, using, and understanding a dot matrix printer. $12.95 plus $2 shipping from MinuteWare, P.O. Box 2392, Columbia, MD 21045 (phone: 301-995-1166).

Epson, Epson, Read All About It! teaches the user the ins and outs of his Epson or Epsonworkalike printer. $14.95 from Addison-Wesley General Publishing Group, Reading, MA 01867 (phone: 617-944-3700).



Timeworks' C-64 spreadsheet, word processor, and database manager stepped up to 128 capability.
READER SERVICE NO. 196

THREE 128 UTILITIES

Timeworks, Inc. has announced enhanced C-128 versions of three C-64 products, each at $69.95:

Word Writer 128 includes an 85,000-word spell checker, a 5-function calculator, and word processing features like document chaining, form letter printout, page separations, and horizontal/vertical scrolling.

Data Manager 128 includes report-writing and label-making capabilities, in addition to search, sort, and chart functions.

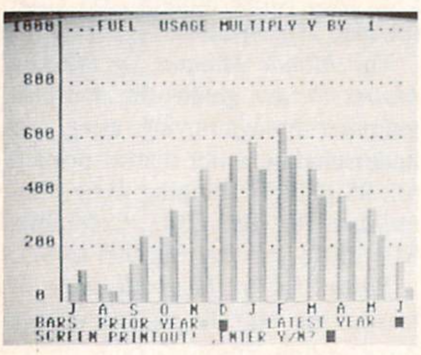
Swiftcalc 128, incorporates the *Sideways* vertical spreadsheet printing utility (see Aug. '85 *Scuttlebutt*).

Timeworks, Inc., 444 Lake Cook Road, Deerfield, IL 60015 (phone: 312-948-9200).

ENERGY USE ANALYZER

Energy Manager lets C-64 users analyze use of heating and cooling energy in homes and buildings by comparing utility bills for the same months in different years. Information is displayed in tabular and bar chart form. Price of $39.95 includes shipping and handling.

Powerline Software, P.O. Box 635, New Hartford, NY 13413 (phone: 315-735-0836).



Analyze energy use in homes, buildings.
READER SERVICE NO. 197

HOME BUDGET SOFTWARE

Plan Ahead consists of three integrated modules, each of which accounts for inflation: *Retirement Planning, Life Insurance Planning,* and *College Funding.* Price of any one module is $29.95, any two $49.95, and all three $59.95.



Lifeplan: *for middle-income families.*
READER SERVICE NO. 198

Sylvia Porter: *first in a series.*
READER SERVICE NO. 199

Advanced Financial Planning, 20922 Paseo Olma, El Toro, CA 92630.

Timeworks has released *Sylvia Porter's Personal Financial Planner* and scheduled for fall release the second in the series—*Sylvia Porter's Personal Investment Manager.* Both for the 64; $59.95 each.

Timeworks, Inc., 444 Lake Cook Road, Deerfield, IL 60015 (phone: 312-948-9200).

Completing the series begun last year, *J.K. Lasser's Your Money Manager* provides a check writer, financial statements, balance sheets, and budget reports. If records are kept accurately all year, they can be downloaded to *J.K. Lasser's Your Income Tax* (1986 version available in December) which will calculate a return in a single step. Price of each C-64 program is $69.95.

Simon & Schuster Electronic Publishing Group, 1230 Ave. of the Americas, New York, NY 10020 (phone: 212-245-6400).

Lifeplan is designed to help middle-income families control and eliminate debts, increase savings, and improve their financial security. A 100% money-back guarantee is included. For the C-64; $99.95.

Lifeware, 2318 Second Ave., Seattle, WA 98121 (phone: 206-322-5454).

SUPER PASCAL

A complete Pascal development system for the C-64 and C-128, *SUPER Pascal* includes a source file editor with renumber, auto, find, and other functions, an assembler for optionally coding in machine language routines, and a comprehensive utility package. The program utilizes a DOS that speeds up 1541 read and write operations by 300%. Included for the $59.95 price is a handbook of over 200 pages.

Abacus Software, 2201 Kalamazoo S.E., P.O. Box 7211, Grand Rapids, MI 49510 (phone: 616-241-5510).

PAPER AIRPLANE MAKER

Why settle for one of the numerous slight simulation programs on the market when your C-64 will now let

you pilot an actual craft? Based on the famous book, *The Great International Paper Airplane Construction Kit* is a set of over a dozen paper airplane templates (to be printed and folded) and a library of aeronautical art for the wings and fuselage, including logos, rockets, and windows. Scheduled for fall release; $24.95.

Simon & Schuster Electronic Publishing Group, 1230 Ave. of the Americas, New York, NY 10020 (phone: 212-245-6400).



The 10' grounding cord of the Model 9250 is fastened to a 3.5' right angle strip extension that folds down and underneath the keyboard. The cord's opposite end should be fastened to a suitable ground. READER SERVICE NO. 200

ANTI-STATIC STRIP

The First Touch Model 9250 Keyboard Strip ($19.95) sticks to your Commodore (or other microcomputer keyboard, where by touching it periodically you help prevent equipment malfunction due to operator-induced static electricity. No, it isn't magic—it's attached to a 10' grounding cord.

3M, P.O. Box 33600, St. Paul, MN 55133-3600 (phone: 612-736-2355).

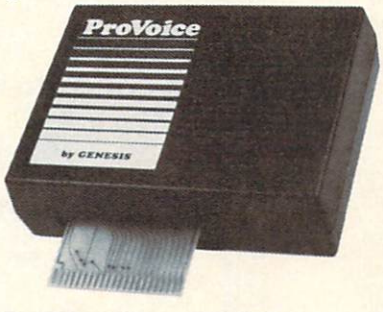
PROVOICE

The second generation of Genesis' COMvoice speech synthesizer, ProVoice features an unlimited English vocabulary, screen echoing (allowing almost any BASIC program to become a talking program), and variable translation modes for conversational, verbatim, and character by character speech output. Thirteen new BASIC commands can be accessed with a single keystroke.

The plug-in device contains a built-in speaker/amplifier. New BASIC commands and text-to-speech translation are handled by onboard ROM, eliminating the need to load additional software.

Price of ProVoice will be $99.95; a complete talking telecommunications package consisting of ProVoice, a 300 baud auto-dial/auto-answer modem, and communications software will sell for under $150. (A Talking Word Processor will be available from Genesis by year-end.)

Genesis Computer Corporation, Ben Franklin Technology Center, Lehigh University, Bethlehem, PA 18015 (phone: 215-861-0850).



ProVoice: 13 new BASIC commands. READER SERVICE NO. 201

MOBILE ROBOT KIT

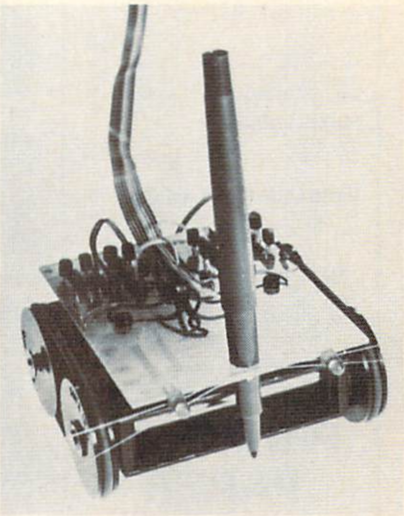
The latest in the Movit family of robot kits (see next issue's feature on *Robotics and the Commodore User*), WAO (Wise, Argent, Orb) includes a microcomputer chip that stores a system program in ROM (2K) and a user program in RAM (128 X 4 bits). The robot can solve high-grade problems and (via the penholder in the center of its mechanism) perform graphics.

OWI Inc., 1160 Mahalo Place, Compton, CA 90220 (phone: 213-638-4732).

Also to be included in next month's robotics feature, the Versatile Mobile Robot Kits from Spectron Instrument include computer interface circuits for the C-64 or VIC 20 that provide remote manipulative output (specify computer when ordering). Pictured is the basic Micro-Kitten.

Base list price is $80 (assembled, $120); with radio control, $195 (assembled, $330). Postage included.

Spectron Instrument, Robot Div., MS 22, 1342 W. Cedar Ave., Denver, CO 80223 (phone: 303-744-7088).



Micro-Kitten: robot for C-64/VIC 20. READER SERVICE NO. 202

BETTER WORKING WORD PROCESSOR

The *Better Working Word Processor* pre-announced in July's *Scuttlebutt* (see page 10) has been released by Spinnaker. The program includes a 50,000-word spellchecker based on the American Heritage Dictionary. For the C-64; $49.95.

Through the Christmas Season, Spinnaker will offer consumers a free Spinnaker product of their choice when they purchase any three of the 50 products offered by Spinnaker's five brands: Spinnaker, Fisher-Price, Telarium, Windham Classics, and Better Working. Details can be found inside appropriately marked packages or with your dealer.

Spinnaker Software Corp., One

Kendall Square, Cambridge, MA 02139 (phone: 617-494-1200).

NEW GAME RELEASES

Infocom's new introductory level fantasy, *Wishbringer,* plunks you down in a town whose locales and inhabitants have disappeared or been changed into sinister new forms—the doing of the Evil One and her henchmen, the Boot Patrol. Worse yet, they've kidnapped a cat, which you must rescue while avoiding the villains. Aiding you is the Wishbringer stone, granting you seven wishes under the proper circumstances. For the 64; $34.95.

Infocom, Inc., 125 Cambridge Park Drive, Cambridge, MA 02140 (phone: 617-492-6000).

Scheduled for fall release are three war simulations for the 64 from MicroProse: *Acrojet, The Advanced Flight Simulator; GunShip, The Daring Attack Helicopter Simulation;* and *Silent Service, The Submarine Simulation.*

MicroProse Software, 120 Lakefront Drive, Hunt Valley, MD 21030 (phone: 301-667-1151).

Two introductory-level strategy contests for the C-64 from SSI, each $39.95.

Colonial Conquest recreates the struggle for world dominance waged in the late 1800's and early 1900's by Germany, France, England, Japan, Russia, and the United States. The computer controls countries not represented by a player.

A solitaire WWII game, *Panzer Grenadier* depicts platoon/company level warfare between the Germans and the Russians on the Eastern Front. Five scenarios are included, each lasting 1-2 hours.

Strategic Simulations Inc., 883 Stierlin Road, Bldg. A-200, Mountain View, CA 94043-1983 (phone: 415-964-1353).

Powerline Casino Roulette allows for most styles of play popular in European and American casinos. On cassette or disk for the C-64; $39.95 (free shipping in Continental US).

Powerline Software, P.O. Box 635, New Hartford, NY 13413 (phone: 315-735-0836).



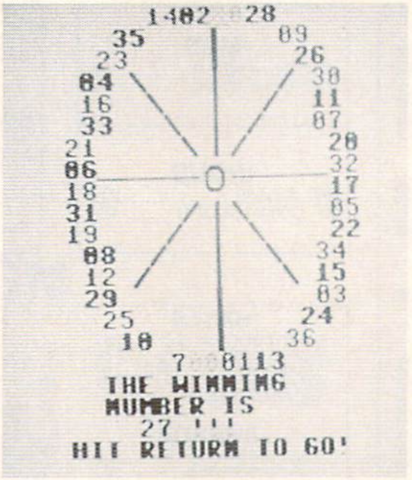
Included with **Wishbringer:** *a map of the town, a sealed special delivery letter, and* **The Legend of Wishbringer,** *a booklet explaining how to use your seven Magick wishes. READER SERVICE NO. 203*

The Wizards have announced that the Cubological Formulation (also known as Rubik's Cube, etc.) will be released in the form of a game for the C-64 by Christmas.

The Wizards, P.O. Box 7118, The Woodlands, TX 77387.

Incorporating 50 animated screens, *Maxwell Manor* requires you to solve the mystery of the previous owner's murder. The man-eating bugs in the garden and the huge spiders downstairs number among your obstacles. For the C-64; $25.00.

Microcomputer Games (division of The Avalon Hill Game Company), 4517 Harford Road, Baltimore, MD 21214 (phone: 301-254-5300).



Onscreen wheel of **Casino Roulette.** *READER SERVICE NO. 204*

EDUCATION PROGRAMS

Ten new C-64/C-128 compatible releases from Learning Technologies:

In the Early Learning Series are five programs for prekindergartners through second graders: *The Flying Carpet* (shape recognition), *Let's Go Fishing* (number recognition), *How to Weigh an Elephant* (ordering and sequencing), *Learning to Add and Subtract,* and *Clowning Around* (visual memory skills).

The Thinking Strategies Series also consists of five programs, these designed to develop problem-solving skills through observation, deduction, and evaluation: *Scrambled Eggs, Sliding Block, Monkey Business, Speedy Delivery,* and *Alpine Train Ride.*

Learning Technologies, Inc., 4255 LBJ, Suite 131, Dallas, TX 75224 (phone: 214-991-4958).

ED SOFTWARE NEWSLETTER

Though the flood of educational software releases have tapered off to a trickle, *The New England SoftNews* will advise users and parents on what's newest and best in home educational programs for children, high schoolers, and adult self-improvers. The newsletter is published six times a year.

New England Software Company, P.O. Box 1154, Kennebunkport, ME 04046 (phone: 207-967-5977).

MIDI PRODUCTS

The latest releases in the burgeoning field of MIDI (Musical Instrument Digital Interface) products for the 64, allowing your computer to interface with MIDI-equipped electronic instruments:

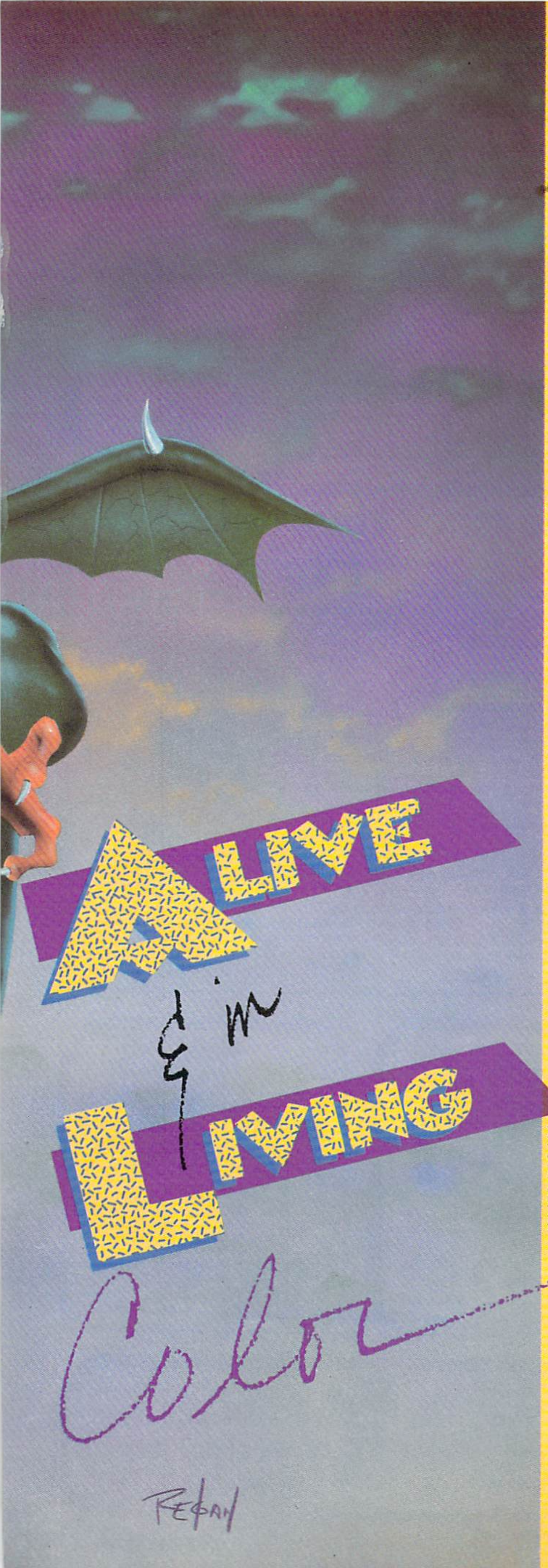
Passport Designs will release a MIDI version of Broderbund's *The Music Shop* for the C-64. A MIDI keyboard, Passport's MIDI interface, joystick, disk drive, and dot matrix printer with graphics interface are required. Price will be $99.95. (See re-

Continued on page 98


SHADOW
THE SHADOW KNOWS
99.9%
EFFECTIVE
Shadow is a new and revolutionary way to duplicate even your most protected software. It encompases all the latest advances in software, as well as a highly sophisticated piece of hardware. This is absolutely the best utility available today. "It will even copy the other copy programs." Because of Shadow's unique abilities, we feel DOS protection is a thing of the past.
By the time you place your order we expect the Shadow to copy 100% — that's right, 100% — of all software available for the C-64.
Order by phone 24 hrs./7 days or send cashier's check/money order payable to Megasoft. Visa, MasterCard include card # and exp. date. Add $3.50 shipping/handling for continental U.S., $5.50 for UPS air. CODs add $7.50, Canada add $10.00 Other foreign orders add $15.00 and remit certified U.S. funds only. Distributors invited and supported.
MegaSoft LTD.
INTRODUCTORY OFFER
$89.95
P.O. Box 1080 Battle Ground, Washington 98604
Phone (206) 687-5116 • BBS 687-5205 After-Hours Computer-to-Computer Ordering
Reader Service No. 218



HOW TO TURN YOUR COMMODORE 64 SPRITES INTO FULL-COLOR ANIMATED CARTOON CHARACTERS—COMPLETE WITH A PROGRAM THAT ANIMATES AND MOVES A DRAGON.



JAMES REGAN

CREATING YOUR OWN GAMES ON THE VIC AND 64

Alive & in Living Color

By Orson Scott Card

Not all action games require realism. Remember when *Asteroids* was new in the arcades? The vector graphics were very abstract—your spaceship consisted of a little wedge, the enemy ships were only slightly more detailed, and the asteroids were nothing but jagged outlines.

At the time, we didn't care. We didn't think it was primitive—it blew us away. To have the little spaceship completely under our control, acting the way it really would in null gravity, with dozens and dozens of things going on at once—I remember crowds standing around the first machine in Orem, Utah, just watching the game.

But we got more sophisticated. The little ghosts in *Pac-Man* with their roving eyes, Mario's padding feet and chugging arms in *Donkey Kong,* the dragons and other odd little creatures in *Dig-Dug,* all led us to expect much more believability, detail, *life* in our games.

Even games that could have been abstract—*Galaxians, Galaga, Firebird*—came ablaze with color and startling, pleasing designs and animation. But there was a limit to what computers could do. Raster graphics couldn't escape from the bumpy look of drawings made out of little square pixels. Animation ate up memory. The more detailed it was, the more memory each shape required. And speed—if your new, lifelike shape used sixteen times as many bytes, then of course it took sixteen times as long to draw it on the screen.

So computer animation became very good—but it reached a limit. Until just a couple of years ago, when they hit us with laser disc games. First-rate cartoonists were going to show us how computer graphics ought to be done, with smooth animation like the best Disney cartoon features.

It should have been nirvana. But it wasn't. And the reason is that once the computer "knows" the shape of a figure, it can "draw" it anywhere on the screen, at any time; but laser disc animation couldn't do that. It was more like a movie. If the hero is two inches from the left side of the screen in frame 1044, then he will *always* be two inches from the left side of the screen in frame 1044.

In other words, while the action became more realistic and lifelike and smooth-moving and pleasing to the eye, the players lost control of the game. You had very little more to do than choose between option A and option B at certain key points. The fluidity, the dexterity of animated games like *Dig-Dug*?

Out of the question for the laser disc.

The power of computer action games is not just in the lifelike animation, though I believe that it is very important. The greatest power is in the fact that the player is in control.

This isn't just a subtle difference. In a movie—animated *or* live—the finished product is the finished product. The audience comes and sees it, and that's it. The audience can't do anything to change the story as it unfolds (except by stamping and screaming and making it impossible to hear, but that's mere aural vandalism).

Now, a computer action game loses the complexity of plot and character and sacrifices much (but not all) of the visual realism. In exchange, however, we players get to create the story ourselves. The gamewright creates a milieu of sights and sounds and gives us certain powers and limitations—you can move here but not there, you can attack this but not that, you have only this many seconds and that many "lives."

Then, within those limitations, and using those powers, we players tell the story ourselves. No, I take it back. We don't tell the story, we live the story, we control a figure that acts it out for us. That's something no other art form can do—let the artist and the audience collaborate in creating related but non-identical stories every time we play.

As the makers of laser disc games discovered, the right balance between realistic animation and player control was struck by the best raster graphics games; the cartoon graphics games were too forced, too predictable, too *uninteresting* for most of us, once the novelty wore off. We were used to being in charge. The laser disc games were like going back to kindergarten.

ANIMATED SPRITES

Which brings us to animating sprites on the Commodore 64. If you've been following this column for the past many months, you'll remember an earlier demonstration of character-graphics animation through character-set flipping (January '85). The sprite shape-flipping technique is closely related. For the sake of newcomers to *Ahoy!*, however, I won't take anything for granted.

Computer animation depends on the same basic technique as cartoon animation. Many pictures are displayed in series for a fraction of a second each. Every picture, called a "frame," is almost identical to the one before, showing only a slight change. In the movies, each frame lasts about a sixteenth of a second. In the old days, the frames came every twelfth of a second—the audience could detect a slight jump or flicker between frames. Today's faster speed has smoothed out the picture completely. Our brain, in the effort to make sense of the image before our eyes, ignores the momentary darkness between frames and "sees" smooth movement.

With sprites, we animate using the same technique. In the example program included with this column (see *Dragon* on page 120), we want to move a dragon around on the screen. We want to create the illusion that the dragon is walking swiftly and breathing fire.

To do this, the program sets up four different shapes for the dragon. Each shape is very similar to the one before it in the sequence. In the shape created using the DATA statements in lines 1020-1031, we see the dragon with its mouth closed, its legs and arms balanced.

The next shape (1040-1051) moves the dragon's farther leg forward, as if he were stepping with his right foot. Also, his mouth opens and flame starts to emerge.

The third shape (1060-1071) evens out the dragon's legs again, just as in the first shape. However, the flames have come farther from the dragon's mouth. The tail is also undulating.

The fourth shape (1080-1091) moves the dragon's nearer leg, as if he were stepping with his left foot. The flames have burst far to the left; the dragon's mouth is now closed.

This sequence is completely circular in suggesting a walking movement. Lewgs even, step right, legs even, step left, and back to the beginning again. The undulation of the tail also completes a cycle. In other words, by repeating the sequence over and over, we can create the illusion of a walking, fire-breathing dragon.

However, the dragon does not always move from right to left. Sometimes he must move from left to right. So we need four more shapes, held in the DATA statements from 1120 to 1191. They are the mirror images of the left-facing shapes from 1020 to 1091.

When the dragon moves to the left, we cycle through the first four shapes, one after the other.

When the dragon moves to the right, we cycle through the last four shapes, one after the other.

SPRITE SHAPE-FLIPPING

In last month's column, we went over sprite creation and movement in detail. If you missed that issue, you can still probably figure out what's going on by consulting the Sprite Memory Map reprinted here, which lists all the key locations for sprite control.

When it comes to animation, the vital locations are the last eight bytes of the 1K block of memory that holds screen memory. When your Commodore 64 powers up, screen memory is at location 1024. That 1K block runs, then, from 1024 to 2047. The last eight bytes are 2040-2047.

These bytes, together, form the Sprite Shape Location Table. The first byte holds the code for the location of the shape for sprite 0; the second byte, the shape for sprite 1; and so on. Since all the sprites are handled similarly, from now on we'll only discuss animating a single sprite, sprite 0.

For the purpose of defining sprite shapes, the VIC-II video chip "thinks" of the 16K video block as being divided into 256 sprite shape blocks consisting of 64 bytes each. Each 64-byte block has its own unique code number, consisting of a number from 0 to 255. The code number for the first sprite shape block in video memory is 0; the code number for the second sprite shape block, the one starting in byte 64, is 1; the code for the last block, the one starting at byte 16320, is 255.

SPRITE MEMORY MAP

For easy reference, here are the sprite-control registers in the Commodore 64. For full explanations, see the text of this month's and last month's articles.

Floating Memory Locations

1016-1023 after the start of screen memory (default 2040-2047): The eight **shape location registers**. (If you're using bit-map mode, they are found at bytes 1016-1023 after the start of bit-map color mem- ory.)

0-63: The codes for the possible location of each sprite shape block within the video block. The actual address is found by multiplying this code by 64 and adding the result to the absolute address of the start of the video block. The first 63 bytes of the sprite shape block determine which dots will be "on." Every 1 bit causes a dot to be displayed; a 0 bit allows the playfield display to show through.

Permanent Control Registers

53248, 53250, 53252, 53254, 53256, 53258, 53260, 53262: Horizontal position registers for sprites 0 through 7, in order.

53249, 53251, 53253, 53255, 53257, 53259, 53261, 53263: Vertical position registers for sprites 0 through 7, in order.

53264: Flag-bit register for horizontal position of all eight sprites.

53269: Sprite enable register. (255 enables all sprites; 0 disables all sprites.)

53271: Sprite vertical expansion register. (255 expands all sprites.)

53272: Screen and character memory location register. Where screen memory is located determines where the sprite shape location registers are located. The high four bits are significant for screen memory location:

screen location=(PEEK(53272)AND 240)/16+*video block starting address*

53275: Sprite priority register. (0 means all sprites appear in front of foreground; 255 means all sprites appear behind foreground.)

53276: Sprite multicolor enable register. (255 makes all sprites display using multicolor option.)

53277: Sprite horizontal expansion register. (255 expands all sprites.)

53278: Sprite/sprite collision register. (0 means no collisions; PEEKing this location clears the register.)

53279: Sprite/foreground collision register. (0 means no collisions; PEEKing this location clears the register.)

53285: Sprite multicolor register 0, controlling the color of 01 bit-pairs.

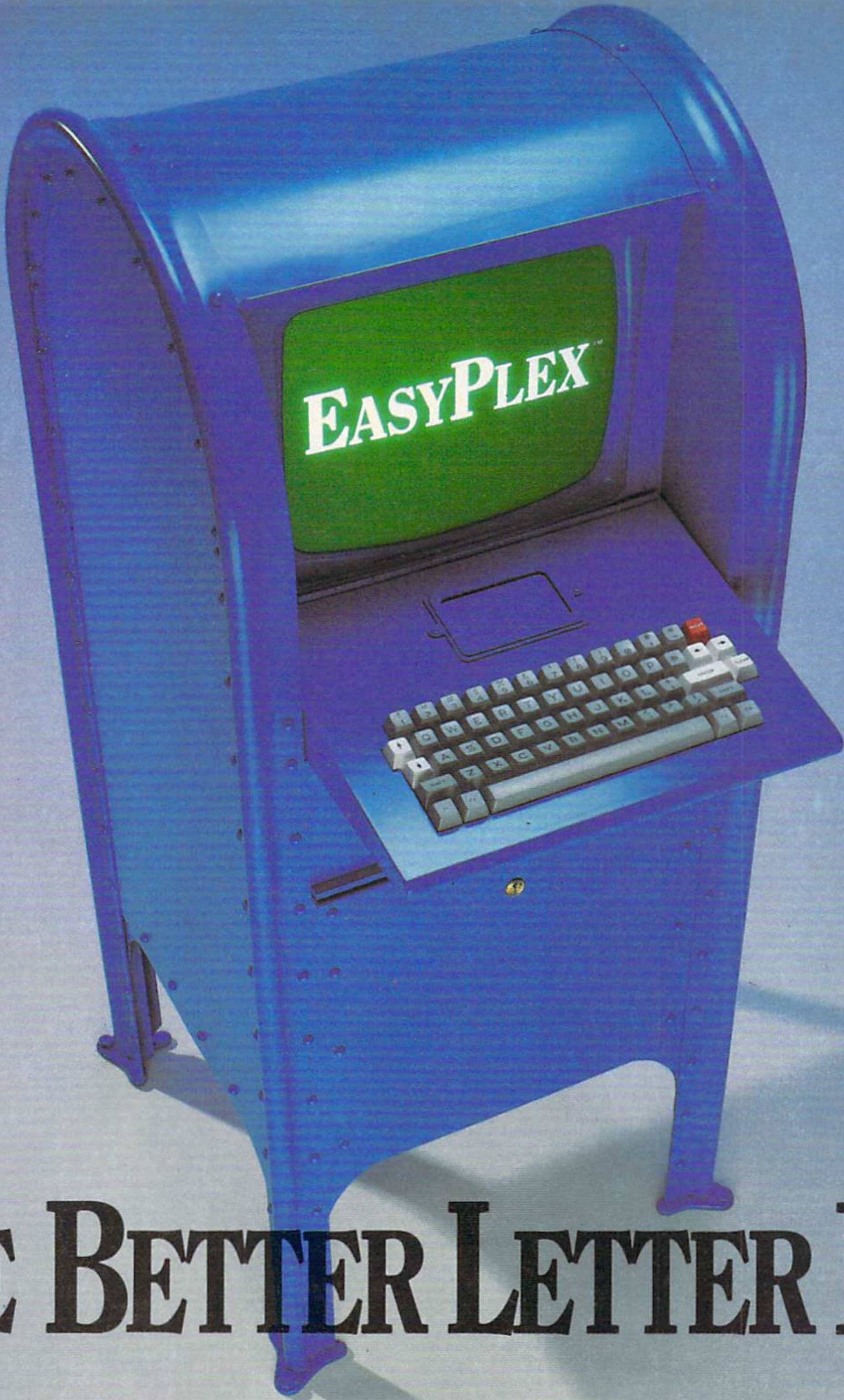
53286: Sprite multicolor register 1, controlling the color of 11 bit-pairs.

53287, 53288, 53289, 53290, 53291, 53292, 53293, 53294: Sprite color registers for sprites 0-7, in order. Default colors are white, red, cyan, purple, green, blue, yellow, and medium gray. (In multicolor mode, these registers control the color of 10 bit-pairs.)

53276: Video block selection register. Bits 0 and 1 determine which video block the VIC-2 chip will read.

To tell the VIC-II chip where to find the shape for sprite 0, you POKE the code number into location 2040 (or the first of the eight bytes of the Sprite Shape Location Table, if screen memory is somewhere else).

Our dragon relies on two separate animation sequences, one for leftward movement, the other to move right. Each sequence consists of four shapes. The routine at lines



THE BETTER LETTER BOX

Introducing EasyPlex.™ The new, easy-to-use electronic mail system from CompuServe.

Finally! Electronic Mail that's so easy to use you can start composing and sending messages the first time you get online.

Designed for various experience levels, EasyPlex has a menu mode with simple, easy-to-follow directions for beginners, and it lets experienced users save time by working in the prompt or command modes. With EasyPlex, you can compose, edit, send, file, and take advantage of sophisticated options previously available only with more expensive services.

CompuServe's EasyPlex lets friends and relatives, associations and club members communicate any time of the day or night. And small business owners, real estate professionals, insurance agents, lawyers, writers, etc. can communicate quickly and simply— either interoffice or interstate. "It's Easy." "Just Plex it!"

Best of all, EasyPlex is available to all CompuServe subscribers. And, along with EasyPlex, you get hundreds of valuable and entertaining computing options. Plus the assurance of belonging to the largest, fastest growing computer information service in the world and the premier supplier of business information to FORTUNE 500 companies.

Start communicating! To buy a CompuServe Subscription Kit, see your nearest computer dealer. To receive our informative brochure or to order direct, call or write:

CompuServe®

Information Services, P.O. Box 20212
5000 Arlington Centre Blvd., Columbus, Ohio 43220
800-848-8199
In Ohio, call 614-457-0802

An H&R Block Company

Reader Service No. 217

ENHANCER 2000



Commodore® Compatible Floppy Disk Drive

One Year Warranty With Second Year Extended Warranty Available

"Where Speed and Quality Meet"

• SUPER HIGH SPEED • SLIM LINE CONSTRUCTION (LOW PROFILE) FULLY COMMODORE COMPATIBLE • DIRECT DRIVE (DC) MOTOR • AVG. MTBF RATE OF 10,000 HOURS • VENTED CASE • SOLID STATE POWER SUPPLY • DUAL SERIAL PORT (CHAINING OPTION) • 5¼" INDUSTRY STANDARD FORMAT

Manufactured Exclusively BY CHINON INDUSTRIES of Japan For "THE COMTEL GROUP"

Foreign And Domestic Inquiries Welcome

1651 East Edinger Suite 209 Santa Ana, CA 92705

Commodore® is a Registered Trademark of Commodore Bus. Machines.

Telephone: (714) 953-6165
Telex: 503727

Reader Service No. 224

1000-1010, after deciphering the DATA statements and converting them into bytes, POKEs the bytes of the sprite shapes into eight sprite shape blocks, the ones with code number 16 through 23.

Now, to display the sprite shapes in order, the main loop in lines 100-196 keeps POKEing different codes into the Sprite Shape Location Table. The variable LT(0) holds the address of sprite 0's shape location register. Line 140, which is executed *every* time the dragon moves, POKEs a different shape code into location LT(0).

If the dragon is moving to the left, line 140 POKEs the codes 16, 17, 18, and 19, and then starts over again with 16. Each time this happens, the VIC-II chip instantly "flips" to the new sprite shape block and displays whatever shape it finds there. If the dragon is moving to the right, the codes 20, 21, 22, and 23 are repeated in an endless cycle.

(The codes, of course, are held in the array ST(XK,AN). XK is either a 0 or a 1, depending on which direction the dragon is facing. AN is a counter variable. Notice that line 100 puts the entire main loop within a FOR-NEXT loop: FOR AN=0 TO 3. Thus, every time the dragon moves, AN changes from 0 to 1, 1 to 2, 2 to 3, and from 3 back to 0. By using AN as an index into the variable ST(XK,AN), we automatically cycle through all four sprite shape blocks for whichever direction the dragon is moving.)

Smoothness—and the Lack of It. When you run the program *Dragon*, you'll find that the animation is not particularly smooth. There is a detectable jump from one shape to the next.

That's because of the speed of BASIC. Every time through the loop, we PEEK location 653, make a few tests, and then POKE three locations. Alas, POKE is a slowish command; it takes time. If the movement lines were omitted, of course, the animation would speed up considerably—but the dragon would just stand in place, which isn't what we had in mind.

Imagine what would happen if we were trying to animate all eight sprites! Each animation step would mean at least one more POKE, and all smoothness would be lost.

If we were using machine language, each change could come very quickly, with plenty of time left over for many other activities. It would be simple to have all eight sprites being animated at once, and each sprite's animation sequences could include dozens of shapes—sequences for movement in four directions, for instance, each of which might include six or seven steps.

Shared Shapes. Since any sprite's register in the Sprite Shape Location Table can point to any of the 256 64-byte blocks in video memory, there's nothing to stop you from having many or all of the sprites point to the *same* shape at the same time. You could easily have eight sprites on the screen, all of them cycling through the same dragon animation sequences. The VIC-II doesn't care whether two sprites are using the same shape block at the same time.

Animation and Movement. Keep in mind that movement and animation are two different things with sprites. Movement is taken care of by POKEing the horizontal and vertical position registers. The dragon *moves* three horizontal dots at a time in this program; it would do this just as smoothly (or jerkily) whether the dragon were *animated* or not. Movement has to do with the sprite's *location* on the screen. Animation has to do with the changing *shape* of the sprite.

MULTICOLOR SPRITES

Besides cycling through animation sequences, you can greatly increase the realism and visual appeal of your sprites by letting them display more than a single color at a time. This is done by "enabling" sprite multicolor mode by turning on the correct bit of the sprite multicolor enable register at 53276.

Bit 0 controls sprite 0. To enable multicolor mode for that sprite, you would POKE 53276,PEEK(53276)OR 1. Bit 7 controls sprite 7. To enable multicolor mode for that sprite, you would POKE 53276,PEEK(53276)OR 128.

If a multicolor mode has been selected for a sprite, the VIC-II looks in the same location to find its shape, but it interprets the shape data differently. In one-color mode, which we discussed last month, the data is arranged like this:

BYTE 0	BYTE 1	BYTE 2
BYTE 3	BYTE 4	BYTE 5
BYTE 6	BYTE 7	. . .
. . .	BYTE 61	BYTE 62

Within each byte, each bit controls one dot on the screen. If the bit is a 1, that dot on the screen displays the sprite's color. If the bit is a 0, then the sprite is "transparent" for that dot, and the regular screen display shows through.

Multicolor mode doesn't see each bit individually. Instead, the VIC-II sees the bits in pairs, four bit-pairs per byte. Each byte, then, looks like this:

00 00 00 00

Each bit-pair, instead of controlling a single dot, controls two dots. Both dots on the screen, side by side, will display the *same* color, whichever color was called for by the bit-pair.

The bit-pair 00 causes the dot-pair to be transparent. The bit-pair 01 causes color 1 to be displayed. The bit-pair 10 causes color 2 to be displayed. The bit-pair 11 causes color 3 to be displayed.

Color 2, displayed by "10" bit-pairs, is individually selected for each sprite. The sprite color registers at 53287-53294 allow you to select any of the 16 possible colors (from 0 to 15) for each sprite.

Color 1, displayed by "01" bit-pairs, is the same for all sprites; that color is set at location 53285. Likewise, color 2, displayed by "02" bit-pairs, is controlled for all sprites at location 53286.

Thus, if all eight sprites were in multicolor mode, and all eight sprites were displaying the same shape, all the 01 bit-pairs in every sprite would display the same color; all the 11 bit-pairs would display the same color; but all the 10 bit-pairs might have different colors. If you plan to have sprites that share the same shape, and you want the player to be able to tell them apart, make sure that the color you want to change from sprite to sprite is represented by 10 bit-pairs.

The dragon in this program is primarily composed of 10 bit-pairs, and sprite 0 is set to green. If another dragon were on the screen, its individual color could be set to blue, and the two dragons would be clearly and obviously different. In a two-player game, this difference in color would let the two players tell their dragons apart. However, if the 10 bit-pairs had been used for the dragon's flame, then the only visible difference between the dragons would come when the dragons were moving and flame was coming out of their mouths; when they were still, with no flame, they would be exactly alike.

HOW THE PROGRAM WORKS

This program, like last month's example programs, reads the sprite shapes from DATA strings. This is not the fastest way to load sprites. The fastest way is to have the sprite data in numeric form in a special disk file and load it with machine language. In BASIC, the fastest way is to have the sprites as numbers in DATA statements.

But for my purposes in these example programs, it's a lot more helpful to give you sprite shapes in a form you can easily edit and change. The idea is for you to type in my dragon program, then change the shapes so the sprite displays whatever shape you want. So instead of speed, I've designed the program for easy editing.

For multicolor sprite shapes, there are twelve bit-pairs in each row of the sprite shape. Therefore, each DATA string consists of twelve characters. Wherever you want the sprite to be transparent, put a period (.) in the string. (Actually, any character except "1" or "2" or "3" would work in the program, but I use the period because it is unobtrusive—but easier to count than blanks.)

Wherever you want color 1 to display, type a 1; for color 2 (10 bit-pairs, remember, the ones individually selectable for each sprite), type a 2. For color 3, type a 3.

The routine at 1000-1010 reads all the strings, converts them into numeric bytes, and POKEs them into memory. There are 21 possible rows in every sprite shape; for the dragon, however, I only used 12 rows. Therefore in line 1000 I set the variable TS to 11 (that is, 12 minus 1). If you want more or fewer rows, make sure you adjust line 1000 accordingly. Likewise, if you want more shapes in your animation sequences, set the variable AS to the number of shapes in your sequence *minus 1. Dragon* uses two animation sequences, one for left movement and one



1541 FLASH!™

The new Skyles Electric Works **1541 Flash!** loads programs and files to your **Commodore 64** *three times faster* than an unenhanced Commodore disk drive can. **1541 Flash!** saves programs 20 to 50 percent faster. **1541 Flash!** formats a diskette in 25 seconds, a *real flash!!*

It gives your 1541 disk drive the speed of much more expensive drives. The Skyles Electric Works **1541 Flash!** is one of the most exciting time-saving devices we've ever seen.

It's Permanent
The **Flash!** is a permanent firmware installation in your **Commodore 64** and 1541 disk drive.

It's Transparent
Best of all, the **1541 Flash!** is transparent. Computer operations all remain unaffected as it speeds up every disk-related function. And there is nothing new to learn for the **Flash!** Once it's in, just watch it go.

We have tested it on well over five hundred programs and it loads spectacularly fast.

It's Easy
Installation of the **1541 Flash!** consists of plugging a small assembly inside the **Commodore 64** and two small assemblies plug into the Commodore 1541. Except for a small percentage of **Commodore 64s**, no soldering is required. Assembly instructions include detailed pictures and drawings. And installation is—well, a flash.

It's Available
Call Skyles Electric Works to place your order or to get more info on the **1541 Flash!** Hurry up, though. They may be gone before you know it.

		FALL SALE
1541 Flash! C-64-128/1541	$109.95	79.95*
1541 Flash! C-64/1541	89.95	59.95*
1541 Flash! with diskswbd	99.00	69.95*
1541 Flash! for two drives	139.95	109.95*
SX-64 Flash!™	99.95	69.95*

128 COMPATIBLE

Storage Power to the People

It's the largest
Yes, the Skyles Electric Works interfaced **Commodore SFD 1001** is the largest floppy disk drive available for the **Commodore 64**. How large? 1.058 million characters of storage. 6.22 times larger than the 1541 Disk Drive. Turn it on, format a diskette and the directory says 4133 blocks free!! No, this is not an expensive hard disk, it is a 5¼ inch, double sided, floppy disk drive. It is the same physical size, shape, color as the Commodore 1541 disk drive. It looks like a 1541, but it stores over 1 million characters of programs and data. Just think of it as 6.22 1541s in one. And at an amazingly low price!

It's the Fastest
Yes, the specially modified **QUICKSILVER**™ interface makes this fastest loading and saving floppy disk drive available for the **Commodore 64**. Over 6 times faster program and file loading than a 1541 disk drive. Twice as fast as a **1541 Flash!**. We made it this fast, because frankly, the advantages of a large disk memory almost disappear unless you have rapid access to the data. And at an amazingly low price!

It's the Least Expensive
Yes, on a price per character of storage basis, the **SEW/SFD 1001** is the least expensive floppy disk drive available for the **Commodore 64**.

		FALL SALE
SEW/SFD 1001 Disk Drive System, C-64	$499.00	299.00*

128 COMPATIBLE

QUICKSILVER™

Is the fastest IEEE-488 Interface available for the **Commodore 64**. How fast? As fast as the external IEEE-488 device will allow. For example, a program or file will load over six times faster from a Commodore 4040 or 8050 disk drive as compared to a 1541 disk drive. **QUICKSILVER** loads 7 times faster than a serial MSD drive. When used with printers speed is only limited by the printer.

QUICKSILVER is transparent. **QUICKSILVER** does not occupy any program memory. Almost all programs are completely compatible with **QUICKSILVER**. A word of caution, some IEEE disk drives have different formats, and 1541 formatted programs are not going to load. Some programs protect the disk by various operations in the 1541 disk memory. These programs will not load on non-1541 disk drives. However, for most uses **QUICKSILVER** is very quick and transparent.

QUICKSILVER is easy. Easy to plug into the cartridge port. Easy to plug additional cartridges into the built-in horizontal, cartridge port extender. Easy to locate your IEEE device within 2 meters (6½ feet) of **QUICKSILVER** using the built-in 2 meter IEEE cable. Easy to select which device numbers you want **QUICKSILVER** to interface to the IEEE bus. Easy to use with almost all programs. Easy to buy, at an amazingly low price, from Skyles Electrics Works or your local dealer.

		FALL SALE
QUICKSILVER, C-64 Cartridge	$129.95	99.95*

128 COMPATIBLE

*There is an additional $4.50 U.S. and Canada, $15.00 Europe and Asia, shipping charge per order. California residents add sales tax.

FOR COMMODORE 128 AND C-64 OWNERS ONLY:

This is just 1 of 15 pages of the newest and biggest Skyles catalog, hot off the press.

We know you'll want this page, in its full 8½ x 11 splendor, and another 14 pages of peripherals, software and books that will make your Commodore 128 or C-64 computer even nicer to live with.

So, if we missed sending you your very own copy within the last few weeks, call us at (800) 227-9998, unless you live in California, in which case call (415) 965-1735.

From Skyles Electric Works, the oldest and largest professional specialists in the business.

Skyles Electric Works
231E South Whisman Road
Mountain View, CA 94041
(415) 965-1735

for right. To have more sequences, change the variable NS in line 1000 to the number of sequences *minus 1.*

The variable MS*(n,n)* is used to hold the possible values for each bit-pair. The values for each bit-pair position are as follows:

BIT-PAIR	POSITION IN BYTE (DECIMAL VALUES)			
00	0	0	0	0
01	64	16	4	1
10	128	32	8	2
11	192	48	12	3

If you want the leftmost bit-pair in a byte to display color 2 (10), you would OR that byte with 128. If you want the bit-pair second from the right to display color 3 (11), you would OR the byte with 12.

The Variables. To help you keep track of what is going on in the program, here's a list of the main variables used in *Dragon.* I deliberately kept it as consistent as possible with last month's programs.

VB	Address of the video block
SB	Address of the screen memory
BB	The screen memory page number (used by BASIC)
LT(0-7)	The address of each sprite's shape location register, located starting at byte 1016 after the start of screen memory.
CT(0-7)	The address of each sprite's color register.
HT(0-7)	The address of each sprite's horizontal position register.
VT(0-7)	The address of each sprite's vertical position register.
HR	The address of the flag-bit register for horizontal movement across the "International Date Line."
ES	The address of the sprite enable register.
VE	The address of the vertical expansion register.
HE	The address of the horizontal expansion register.
PR	The address of the priority register.
EM	The address of the multicolor enable register.
MR	The address of the first multicolor color-selection register.
CS	The address of the sprite/sprite collision register.
CF	The address of the sprite/foreground collision register.
BS(0-7)	The bit-set values for each sprite.
BC(0-7)	The bit-clear values for each sprite.
MS(0-3,0-3)	The bit-pair values for multicolor byte selection. The first index determines which bit-pair is called for (numbered from *right* to *left*), and the second index determines which color is being called for.
ST*(n,n)*	The codes for the location of each sprite shape block; these numbers are POKEd into the sprite 0 location register LT(0). The first index determines which animation sequence is being used, left or right; the second index determines which step in the four-shape sequence is to be displayed.
NS	The number of sprite animation sequences, minus 1.
AS	The number of shapes per animation sequence, minus 1.
TS	The number of three-byte (12 bit-pair) lines per sprite shape.
VV*(n)*	The vertical position array; these values are POKEd into the vertical position registers at VT*(n)*.
HH*(n)*	The horizontal position array; these values are POKEd into the horizontal position registers at HT*(n)*.
HB*(n)*	The horizontal flag-bit array; these values are POKEd into the flag-bit register at HR.
VZ	The lowest vertical position allowable on the screen, minus 1.
HZ	The rightmost horizontal position allowable on the screen, minus 1.
VP	The current vertical position (a number from 0 to VZ).
HP	The current horizontal position (a number from 0 to HZ).
AN	The current step in the animation sequence.
XP	The direction the dragon is currently facing.
Q	A flag. If it is 1, it means the dragon has been moved since it was last at rest; if it is 0, it means the dragon is at rest and has not been moved. It is used to determine whether the dragon needs to be set back to "resting" position—with no flame showing.

Starting and Stopping. The program relocates video memory and disables the RUN-STOP/RESTORE, and SHIFT/COMMODORE interrupts. If an error interrupts the program, you will probably have to turn off the machine and reload the program. So *please* SAVE the program *before* you RUN it, since the slightest typo can crash the program and make it very hard to recover it.

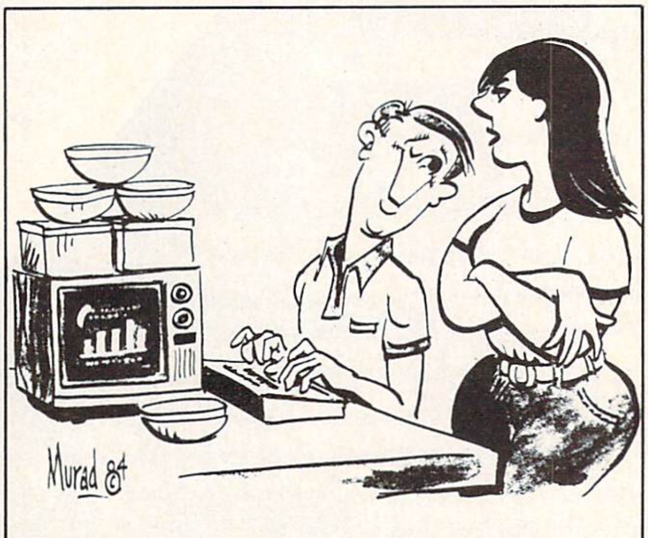
To stop the program once it is running, press SHIFT/COMMODORE/CTRL. This restores all the video pointers and reenables the interrupts.

To move the dragon to the left, press COMMODORE. To move the dragon to the right, press SHIFT.

NEXT MONTH

Next month we'll conclude this series on sprites with a program that creates a spaceship that can move in eight different directions, along with a few celestial objects. The BASIC version of the program will be quite slow, unfortunately; if I'm feeling particularly ambitious I'll create a machine language interrupt routine to read the keyboard and move the spaceship, in addition to the BASIC movement method. ☐

SEE PROGRAM LISTING ON PAGE 120



"Let me get this straight...your bargain-brand computer doesn't come with any software, but it does come with a lot of Tupperware?"

MYSTERY AT MYCROFT MEWS



for the C-64

By John R. Prager

Scandal has rocked the small village of Mycroft Mews: a prominent citizen has been callously murdered for reasons unknown. Your job as a detective is to ensure that justice triumphs. It won't be easy. The bank president won't talk. The hotel clerk's alibi is clearly bogus. The vicar's fingerprints are on the elephant gun you found at the Pawnshop. Any of these shifty characters could be the killer!

Mystery at Mycroft Mews is a game for the Commodore 64 that requires all the logic and cunning you can muster. Competing against you will be as many as five other detectives; you'll have to outfox them to solve the mystery first. There are nearly 5000 possible solutions, but the single correct answer is chosen anew each time the program runs. To crack the case, you must correctly deduce this unique answer before the other players do.

To play, type in the program from the listing on page 127 exactly as it is shown. It is especially important that you do not alter the DATA statements which appear from line 30000 onward. If you wish to use a joystick, you may insert it into either port. The game accommodates from one to six players, who can compete as individuals or as teams of equal size. You may want to supply pencils and paper for each player to use in taking notes.

MOVING THROUGH TOWN

As each player begins his or her turn, all other players should position themselves so they cannot see the screen. After all, one goal of the game is to acquire information that other players do not have. To help preserve secrecy, you will have to type in a password at the start of your turn whenever two or more players are competing. This password can be any two characters from the keyboard...letters, numerals, or even special keystrokes such as the function keys. Only after you type your code will you be allowed to continue with your turn. This prevents other players from illegally using your turn to gather information for themselves.

After you type in your code, you may travel to any of the 16 buildings in *Mycroft Mews*. A sprite "detective" will appear on the screen; he moves from building to building on the map under joystick control. If you wish to play without a joystick, use function keys f1, f3, f5, and f7 to move, respectively, North (up), West (left), East (right), and South (down). To enter a building, position


EXPAND IT
NAVARONE
NOW FOR YOUR COMMODORE 64
EXPAND YOUR COMMODORE 64™ WITH NAVARONE'S NEW CARTRIDGE EXPANDER
Plug in up to three cartridges at a time. Select the program you want. Even reset the computer independently of the power switch.
• Reduces cartridge swapping
• Built in reset button
• Saves wear & tear
NOW ONLY $24.95
NAVARONE
Navarone Industries, Inc.
19968 El Ray Lane
Sonora, CA 95370
(209) 533-8349
Reader Service No. 226
*Commodore 64 is a trademark of Commodore Electronics, LTD. respectively. This product is manufactured by Navarone Industries for use on the Commodore 64 computer and is not made, sponsored, authorized or approved by Commodore Electronics, LTD.

the detective immediately to the right of the door and press the joystick button or RETURN key. Of course, if you start your turn in a building, you may remain there for as long as you wish, merely by pressing the button instead of moving.

One of the buildings – the home of the victim – has been locked by the police at the start of the game. It can be entered only by a player who carries the Master Key, which is originally located at the Police Station. The map of the village changes every time the game is run, so finding the Police Station is a challenge in itself. Be sure to note the nature of every building you enter!

After you enter a building, the computer will describe your surroundings. It tells you what building you occupy; notes what object you carry; lists all suspects, weapons, and useful items in view; and names any other players that are present.

TURN OPTIONS

Following this description, you may press the joystick button or RETURN key for a list of options for your turn. The menu of options will vary, depending on your location and the items immediately available. To choose from any menu, use the joystick or function keys to position the highlighted bar at your choice, and press the joystick button or the RETURN key.

The SEARCH option is available every turn. Choose it if you believe an object may be hidden in the building you occupy. Initially, all 12 of the possible murder weapons are hidden in one building or another. Each time a player searches a given area, there is a 60% chance of finding each item hidden there. For this reason, a location may have to be searched repeatedly before all the objects hidden there have been found. The Magnifying Lens, initially located at the Police Station, allows the player carrying it to search with 100% effectiveness. A successful search gives you the choice of taking the object that you have found.

Choose the TAKE option to pick up a weapon or useful item in the area you occupy. Since you may carry only one item at any time, you will drop any object that you carry when you take another. If you select this option, you may even take objects that other players possess. The DROP option allows you to get rid of an object that you no longer wish to carry. If you so choose, you may even hide the object that you drop, so that other players will have difficulty tracking down specific clues.

Certain optioins are limited to specific locations. Only at the Morgue can you choose to READ THE CORONER'S REPORT, which will reveal the gender of the murder victim and provide a hint about the weapon used. At the Police Station, you may FINGERPRINT an item to determine its owner, or you may ACCUSE a suspect in hopes of winning the game. At the Bank, Law Office, and Mental Health Clinic, EXAMINE THE RECORDS about a suspect to discover useful facts about the motive behind the crime.

Hidden somewhere in the victim's house is a vital piece of evidence: a will, diary, or ledger. Find it. READ it. You will learn the motive behind the slaying and obtain a list of suspects that includes the murderer. Try to hide this clue where the other detectives won't find it.

GRILLING SUSPECTS

At the heart of the game are the suspects: the dozen leading citizens of Mycroft Mews. You will encounter only eleven of them in the course of the game, for the twelfth was the victim. One of the survivors is the killer. Although each suspect has a particular building as his or her "home base," they move about from turn to turn, so you can encounter anyone in any location except the victim's house. A suspect will return to his home building, however, on the turn after you arrive there.

If you are in a location with one or more suspects, you may choose the INTERROGATE option. If several suspects are present, you will be asked to choose one to question. You may ask three questions of your informant by moving the highlighted bar to the subject you want to know more about. Press the joystick button or RETURN key to ask about the item or suspect currently highlighted. If you move to the bottom line of the menu and press the joystick button or RETURN key, you will be presented with a new menu of topics for your inquiries; changing menus in this manner is not counted as one of your three questions.

Interrogating will help you enormously by revealing alibis, motives, and interests of the suspects. Remember, though, that the character you are questioning answers on the basis of rumor and hearsay, and his answers are not always reliable. Often, too, suspects are too busy to gossip with a nosy detective...although they will not refuse to answer if you possess the Police Badge, which lends an air of authority to your investigations. Like the other two useful items, the badge is originally found at the Police Station.

Most important, the murderer himself is free to lie. He will always deny ownership of the murder weapon, even though his fingerprints are on it! For this reason, it is vital that you jot down not only the testimony that you gather, but also the source who disclosed it.

WINNING THE GAME

When you are certain that you have all the pertinent facts at hand, go to the Police Station and choose the ACCUSE option. You will need to identify the victim, the murderer, the weapon used, and the motive behind the crime. If your answer is not correct, you are eliminated from the game. A correct solution wins! If all players have made incorrect guesses, or if the case if solved, the correct answer will be displayed.

Finally, to cut down on trips to the dictionary: an "arquebus" is an antique form of rifle that is greatly prized by gun collectors, and a "herpetologist" specializes in the study of reptiles, and might well have a vial of snake venom on hand. Happy sleuthing! □

SEE PROGRAM LISTING ON PAGE 127

GRAVINAUTS

for the C-64

BY CLEVELAND M. BLAKEMORE

On the world of Arkion, survival is a tough business. Because the surface of the planet is covered by a deadly acidic ocean,the inhabitants must remain continuously in the air, supported by rocket backpacks. To stay up, they engage in a never-ending war for fuel. The loser is doomed to a very short lifespan.

The Gravinauts, as they are called by the colonies of the Milky Way, battle each other by trying to reach a higher vantage point in the air than their opponents and smashing into them, destabilizing their gyros and causing their backpacks to explode.

Because the fuel tanks stored higher up in the atmosphere are more closely guarded, they are entrusted with greater fuel supplies. They are therefore more important than the ones resting on the lower "ice glaciers," which are huge bodies of frozen chemicals floating in the air above the ocean.

The purpose of the game is to take fuel off the glaciers, destroy your opponents for points, and advance to the next level. The game gets increasingly more difficult at each level, as the speed of the game increases, and at the highest level your visibility is affected when a fog bank of chemicals sets in, obscuring your view of the glaciers.

You must conserve fuel, using rebounds off objects as much as possible, and try to knock out your enemies with as little effort as is necessary. If you run out of fuel, your rocket backpack will explode.

The game is played with a joystick in control port 2. The controls are simple. Push the fire button with the joystick centered to fire a straight thrust of exhaust, push the fire button with the joystick to the left to propel yourself left, and with the joystick to the right to go right. These are the only controls!

Gravity is the only thing that will make you sink, so if you fire too much fuel you may begin gyrating out of control, rebounding off glaciers until you can stabilize yourself. Because you are in "Zero G" gravity, there is no loss of kinetic energy when you bounce off something–so you could keep bouncing *ad infinitum* until you slow yourself down with short bursts of fuel.

Each enemy gravinaut is worth 250 points–you may occasionally get a double score of 500 off one opponent. The fuel tanks contain fuel supplies in increments of five from top to bottom, with the tank at the lowest level worth five and the highest worth twenty. If you manage to destroy all three of your opponents, and pick up all the tanks of fuel, the screen color will change and you advance to the next level. Each time you start out on your platform at the bottom, the only safe spot on the sea of acid. Any contact with this solution at the bottom will destroy your player. You can also score points by leading your opponents into the acidic sea, where they will be quickly burned up.

You only have one man per game. If he blows up, push f1 to start a new game at level one.

There are situations where you may have to hit an object solidly for it to be affected. Sometimes you can take a glancing blow on the surface of the acid sea or from one of your opponents without being killed.

High score since the game was started is displayed in light blue in the upper right corner. My highest so far is 7550, which you will have to beat by passing through several of the upper level "invisible" screens, where the screen is the same color as the glaciers and they cannot be seen.

The game is typed in as a hex listing. Be sure to use *Flankspeed* (see page 118) when typing it in so that you can be assured of an error free program. □

SEE PROGRAM LISTING ON PAGE 143

COMMODORE COMPATIBLE SINGLE DISK DRIVE



Commodore™ owners, are you ready for a disk drive that delivers more FEATURES, PERFORMANCE AND COMPATIBILITY at a competitive price to the 1541? Peripheral Systems of America CS-1™ gives you all that the existing drives offer and much more.

FEATURES:

- 100% compatible.
- Reset button to save wear and tear on your disk drive.
- Free utility software - Q-Load (fast load), Copy-Q (fast copy).
- External switch for selecting device number.
- Data error detection and correction feature.
- Reliable @ Save function.
- Formats in 16 seconds.
- State of the art design - streamlined with an external power supply.
- Reliable and durable.
- 90 day limited warranty.
- 9 month extended warranty available.

C-64 is a registered trademark of Commodore Business Machines Inc.
DEALER INQUIRIES WELCOME

Other products offered by Peripheral Systems of America:

FOR COMMODORE

Hardware: Dual Drive, Serial Interface, 300 Baud Modem, Graphic Printer Interface, Fast Load/DOS Cartridge, Light Pens, Joysticks.

Software: Terminal Program, Back-Up Copy, Five Modular Accounting Packages

OTHER COMPUTERS

4-Color Centronics Plotter/Printer, Disk Notcher

PSA

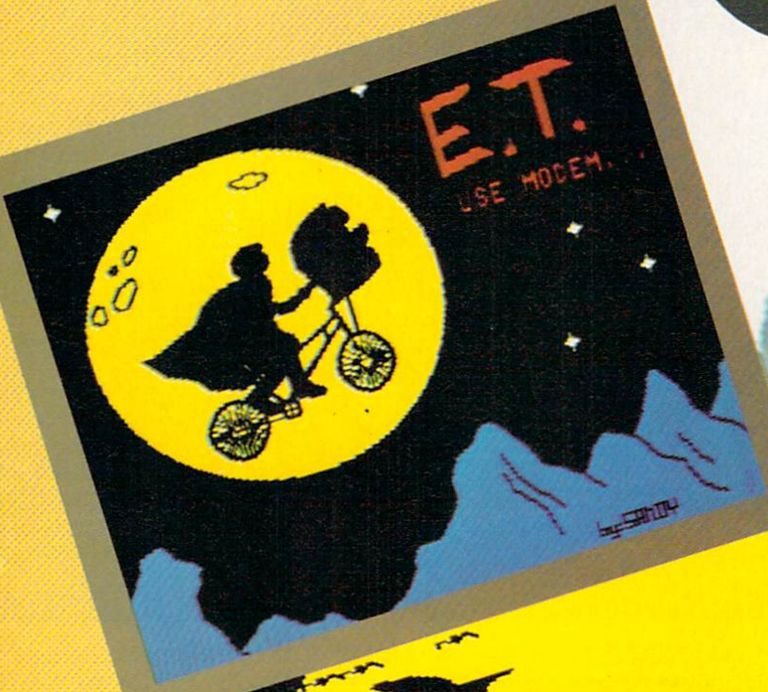
Peripheral Systems of America, Inc.

2526 Manana Rd., Suite 209, Dallas, TX 75220 (214) 956-7866

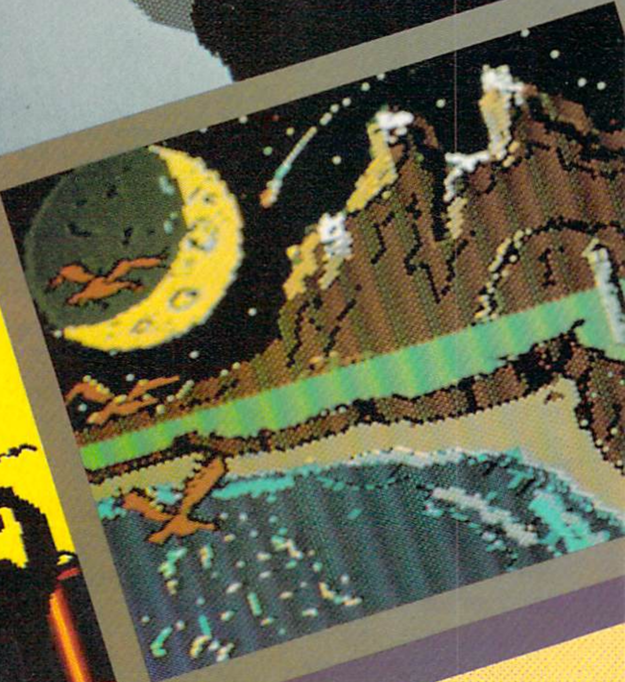
PRODUCTS YOU CAN BELIEVE IN

Reader Service No. 228

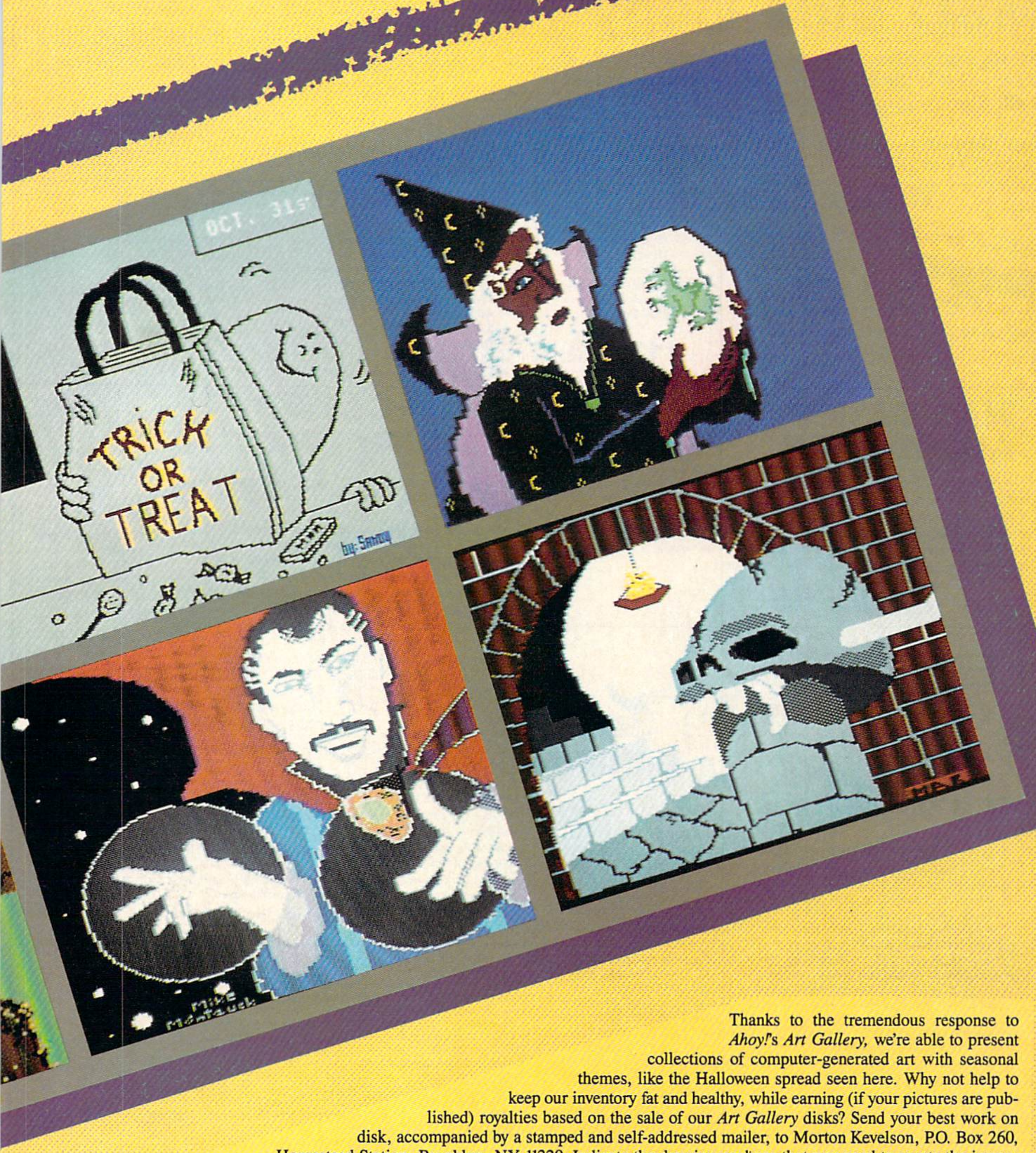
ART GALLERY











Thanks to the tremendous response to *Ahoy!*'s *Art Gallery,* we're able to present collections of computer-generated art with seasonal themes, like the Halloween spread seen here. Why not help to keep our inventory fat and healthy, while earning (if your pictures are published) royalties based on the sale of our *Art Gallery* disks? Send your best work on disk, accompanied by a stamped and self-addressed mailer, to Morton Kevelson, P.O. Box 260, Homestead Station, Brooklyn, NY 11229. Indicate the drawing package that was used to create the image. If you employed a bit map of your own design, indicate the appropriate file parameters, i.e., hi-res or multicolor, location of bit map, screen or color data.

Top row, left to right: *E.T.* by Sandra Steele (Rockford, IL), drawn on *DOODLE!; Our Cat,* a *Koala* rendering by Mark Richey (Boulder City, NV); *Trick or Treat,* another *DOODLE!* from Sandra Steele; and *Wizard* by David Moroz-Henry (Downington, PA). This last was done completely using programmable characters – no mean feat. The dragon inside the crystal ball is actually composed of four sprites. When the program is run, the wizard appears first. A few seconds later the dragon appears inside the crystal ball. Bottom row, left to right (all drawn on *Koala*): *Witch* by Michael Montauck (Brooklyn, NY); *Nightfall* by Don Cadle (Riverside, CA); *Dr. Strange* by Michael Montauck; and *Skull* by Mark Richey.

MSD Sure Copy

At last a complete utility package for the MSD Dual Drive. **This is the first MSD utility program that does it all.** The main menu options include:

Sure Copy will put all errors automatically on disk: 20, 21, 22, 23, 27 and 29's.

- Copy Protected Disk
- Copy Files
- Format a Disk
- Change Disk Name
- Quit
- Copy Unprotected Disk
- Scratch a File
- Rename a File
- View Directory

$39.95

TOP SECRET STUFF I



- DMS
- ½ Track Reader
- ½ Track Formatter
- Drive Mon
- The Doc
- Sync Maker
- Sync Reader
- Change Drive No.
- Disk Logger
- Disk Match
- New Wedge
- ID Check
- Unscratch
- View RAM
- Read/Write Test
- Repair a Track
- Fast Format

$19.95

TOP SECRET STUFF II

All C128 Compatible

- Split Screens (Horizontally)
- Smooth Scrolling
- Save Ram From Under Roms
- No Drive Rattle On Errors
- Triple Drive Head Speed
- Autoboot Maker
- Koala Screen Dump
- Display GCR
- Fast Disk Eraser
- Protect Scheme For Your Disks
- Write Protect Disk
- Unwrite Protect Disk
- Mini D.O.S. Wedge
- Fast Diskmatcher
- Data Statement Maker
- Unnew
- 3 Minute Copy
- D.M.S.

$19.95

WAR GAMES AUTODIALER



1-Auto Dial will automatically dial a set of numbers you choose.
2-Review Numbers will review numbers that were answered by a computer.
3-Save Numbers will save numbers where a computer answered.
4-Hardcopy of Numbers will print out list of numbers where a computer answered.
5-LOAD Numbers will load in numbers to continue where it left off.
6-Continue will pick up dialing where it was interrupted.

$29.95

D-CODER

- Translates any machine language program into easy-to-read English descriptions with complete explanations of each command!
- Makes complete notations of all important memory locations accessed by the program! (SID, VIC, MOS, KERNAL, etc.)
- Gives you three ways of accessing programs:
  1) **Will read and list programs from DISK**
  2) **Will read and list programs from MEMORY**
  3) **Direct user imput** (from magazines, etc.)
- Can be used to locate and examine any machine language program's protection routines!
- Can be used to easily break apart machine language programs for study and examination!
- Printer option for complete hard copy listings!



You Now Longer need to be an EGGHEAD to read Machine Language.

$19.95

SOFTWARE PROTECTION HANDBOOK

3rd Edition

THIRD EDITION! NOW AVAILABLE!

If you're **tired** of being harassed by **protected software** and too many **copy** programs, then this is the book for you! This 224 page manual covers the gambit from legalities to protection methods to step-by-step back up procedures. Now you can learn both how to **protect and unprotect** software! The techniques covered include copying cartridges to tape or disk, tape protection, and disk protection. Disk protection covers error no.'s 20, 21, 22, 23, 27 and 29 **plus** single track formatting, header modification, header swapping, half track reading and writing, reading and writing modified bit densities, formatting illegal tracks/sectors, sync writing and **more!** The Third edition **explains,** tells how to **detect** and how to **write** them with included software. Eleven useful utilities and many protection listings! Our disk analysis programs reveal the protection methods used on your originals. A diskette with all software is available for a minimum extra charge. This may not be the only book your should have for the C-64, but it is certainly **the one** book you should not be **without!**

C64 Book only	$19.95 US
Book & Disk of all programs	$29.95 US
Vic 20 book . . . Cart. & Tapes only	$9.95 US

THIS MANUAL DOES NOT CONDONE PIRACY

*SHIPPING: $2.00

N-CODER

the machine language manipulator...

The perfect companion program to D-CODER!

Allows you to easily make changes in machine language programs... right on the disk!

- Rewrite ability allows code to be altered and then rewritten directly to the disk!
- Features sector-by-sector scrolling assembly language display of machine language programs!
- Notation of ASCII text equivalents for easy spotting of embedded text strings!
- handy reference display of all assembly language commands and their ML numerical equivelents!
- Byte splitter for easy splitting of decimal addresses into low byte-high byte format!

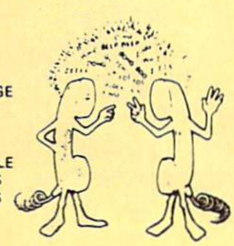


$19.95

Bulletin Board

Set up and operate your own bulletin board with one or two disk drives. This one has all the features and you can customize it easily yourself.

1-RUN MEGASOFT-BBS
2-CREATE MEGA FILES
3-ADD TO SYSOP'S CORNER
4-NEW SYSOP'S CORNER
5-READ MESSAGES
6-SCRATCH MESSAGE
7-CYCLE MESSAGES
8-READ SYSOP MESSAGES
9—WRITE OPENING MESSAGE
10-READ LOG
11-CYCLE LOG
12-READ DOWNLOAD FILE
13-SCRATCH DOWNLOAD FILE
14-ADD TO OTHER SYSTEMS
15-CREATE OTHER SYSTEMS



$59.95 **BBS**

- Leveled Access
- Private Message Base
- Up to 300 Passwords
- Expert Mode
- Open Chalk Board for Highest Level Access

Enclose Cashiers Check, Money Order, or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders. Canada orders must be in U.S. Dollars. VISA — MASTER CARD — C.O.D.

Programs for C-64 $2.00 S & H on all orders

Software Submissions Invited.

MegaSoft Limited

P.O. Box 1080, Battle Ground, Washington 98604

Phone (206) 687-5116 • BBS 687-5205 After Hours Computer-to Computer

Reader Service No. 219

SOLITAIRE 64

By Eric W. Sink

Almost everyone has lost at solitaire. This computer version is your chance to get even. Sure, you'll still lose, but when you get really frustrated, you can just turn the computer off.

I'm not going to go into the rules of the standard game. If you don't know how to play solitaire, someone around you certainly does.

Type the game in and save it immediately. Then run it. A title screen will appear and disappear, followed shortly by the game selection screen. *Solitaire* will ask you if you wish to play Las Vegas style or regular. For now, choose regular by responding with a joystick in port 2. I'll explain Las Vegas style later.

A graphic display of your "table" will appear, complete with your beginning cards. All the blue numbers above the seven piles indicate the number of cards that are still hidden from your view. The blue numbers above the two stacks of the deck at the bottom indicate how many cards are in each stack.

Notice the blue arrow in the middle of the screen. This is your pointer. All facets of gameplay are controlled by moving this pointer with your joystick. To move a card from anywhere to anywhere else (assuming it's a legal move!), move the pointer, with the joystick, to the card you wish to move. Press the fire button to pick the card up. The card to be moved will turn blue. Now move the pointer to the card's destination. Press the button again to drop the card and the move will be executed. If the move is illegal, the computer will buzz at you and give you a chance to try again. If you make a mistake and pick up the wrong card, simply drop it right back where it came from.

To move more than one card at a time, pick up the first card in the group you wish to move. For example, if you wished to move a King and all the cards on top of it to a blank pile, simply pick up the King, and the computer will assume you wish to pick up all the cards on top of it. You may move only one card at a time up to the aces.

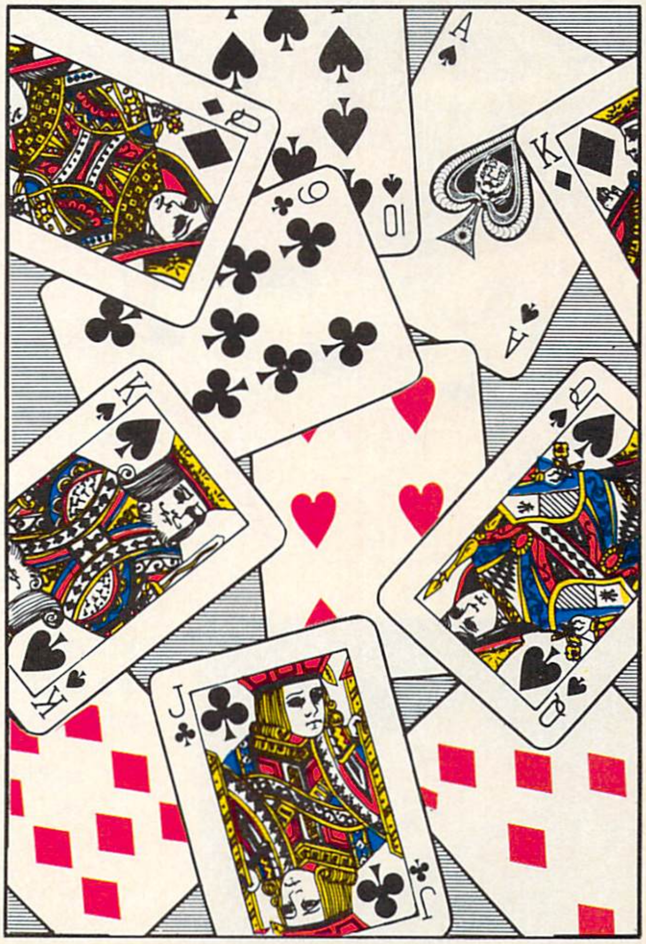
Very soon, this process of moving your pointer around will become second nature to you. When it does, here are a couple of shortcuts you will probably have figured out on your own. First, to drop a card on the aces at the top of the screen, simply move the pointer anywhere above the blue numbers and drop it. The computer will put the card where it belongs, since it can have only one destination. Second, when dropping cards to a pile, vertical positioning doesn't matter as long as you are below the blue numbers (at top) and above the top of the deck (at bottom). To turn up the next three cards, move the pointer to the deck and press the button. When you get to the bottom of the deck, the words "Turn deck over" will appear in place of it. Press the fire button on these words to do just that. Press the fire button on the word "Concede" at the bottom right to quit a game.

The computer is the judge in this game. It will not allow you to cheat. It will play out the rest of the game if it sees that you cannot lose. It will never tell you that you have lost. You get to keep trying until you concede.

Now a word about Las Vegas style. In this variation, you buy the deck for $50. You turn up one card at a time instead of three, but you may only go through the deck once. You get $5 for each card that is in the aces section at the top. At the end, the computer will tell you how much you won or lost. Everything else is the same as the regular version.

I hope you enjoy playing the game, and may the luck of the cards be with you! □

SEE PROGRAM LISTING ON PAGE 123





YOUR MISSION: TO EXPLORE NEW WORLDS

Tap into an exciting computer resource available for the VIC-20 and C64 computers: synthesized speech.

Talktronics brings to you two extremely powerful plug-in speech synthesizers to let your computer sound off in ways you never knew were possible. **VIC-Talker** and **64-Talker** enhance your educational programs and games, aid the handicapped and generally increase computer "friendliness".

Look at these sophisticated features:

- Self contained firmware – disk or tape not required for synthesis.
- Audio mixer to combine computer generated sounds with the synthesized voice for **VIC-Talker** or **64-Talker.**
- Two user-programmable voices that can be as varied as your imagination – from chipmunks to robot warriors.
- Translation of ordinary text directly to synthesized speech using pronunciation rules for English.
- Nonvolatile user-programmable exception word memory permits special pronunciation or translation of words you select and is retained when power is off.
- Contains rules for pronunciation of text, numbers, punctuation and BASIC command words.
- Accurate translation of number strings in decimal, dollars and cents, fractions, BASIC equations, time and phone numbers.
- Easily controlled with simple commands from BASIC or machine language – like using the printer or disk.
- **Proofread** mode announces the punctuation in a different voice – lets your computer "speak" BASIC program listings.
- Change the voices along with speaking text. Make it sing!
- The voice comes out on the TV or monitor.



VIC-Talker and 64-Talker are priced at $89 each and are available from

Talktronics, Inc.
27341 Eastridge Drive
El Toro, CA 92630

(714) 768-4220
9 AM - 5 PM, PST



Reader Service No. 222

MAESTRO

An Advanced Music Editor for the C-64

By William T. Eveleth

This extremely versatile music editor makes playing music on the Commodore 64 as easy as Do-Re-Mi. If you think word processors make writing easy, wait till you try your hand at *Maestro.* You can create original music for up to three independently controlled voices with no more trouble than typing a letter. And that's not all. *Maestro* lets you make changes as you go. Change a wrong note. Change the rhythm–or, if you like, change the tempo. Type in your favorite song from the sheet music. Or, create your own melodies. You be the composer–with an electronic orchestra at your fingertips! And once you've completed your masterpiece, you can save it on disk for a future premiere concert.

Maestro is organized into 10 operating modes as follows:

MODE 0: MUSICAL KEYBOARD

In this mode your keyboard becomes the musical instrument of your choice. Every key produces a sound (except RESTORE, SPACE, CONTROL, and the COMMODORE key). The corresponding notes on the piano are shown in the illustration on page 40. The same notes are shown in the staffs above each keyboard section. To switch instruments, type SHIFT to return to the menu, select MODE 5, make your choice, and return to MODE 0. (Note: not all 10 instruments can be played from the keyboard in this mode.)

MODE 1: PLAY IT AGAIN!

This mode is similar to MODE 0, but with one major difference: *Maestro* remembers what you played! Type SHIFT after playing a few notes, and *Maestro* will play it back. Exactly the way you played it–the same notes, rhythm, and tempo.

To leave this mode before the playback cycle is completed, press the COMMODORE key.

MODE 2: COMPOSE A TUNE

Use this mode to create an original composition or to enter notes from sheet music. First indicate the number of voices. Think of a voice as one musical instrument. With only one voice, we are limited to the melody. If we wish to add harmony, additional voices are required. Most music is written for several voices, and the SID chip can handle up to three. With a little experimentation, you will find that the more voices, the fuller the music sounds. If you're trying to input notes from sheet music, you will probably find that more than three voices are needed. For example, five voices would be required to play a five-note chord. However, if three notes are carefully chosen from such a chord, the same effect can usually be achieved.

Once you've indicated how many voices will be needed (and it's always possible to add or delete voices later), you will start to input the notes to be played by each voice. Using Figure 1 or 2 as a guide, type the notes in the correct order. Be sure to include rests (pauses) by typing SPACEs. If you forget a rest, the rhythm will probably be incorrect.

After you have input a series of notes, type SHIFT, and you will be prompted for the corresponding rhythm for each note. For each note, you must indicate the note's duration.

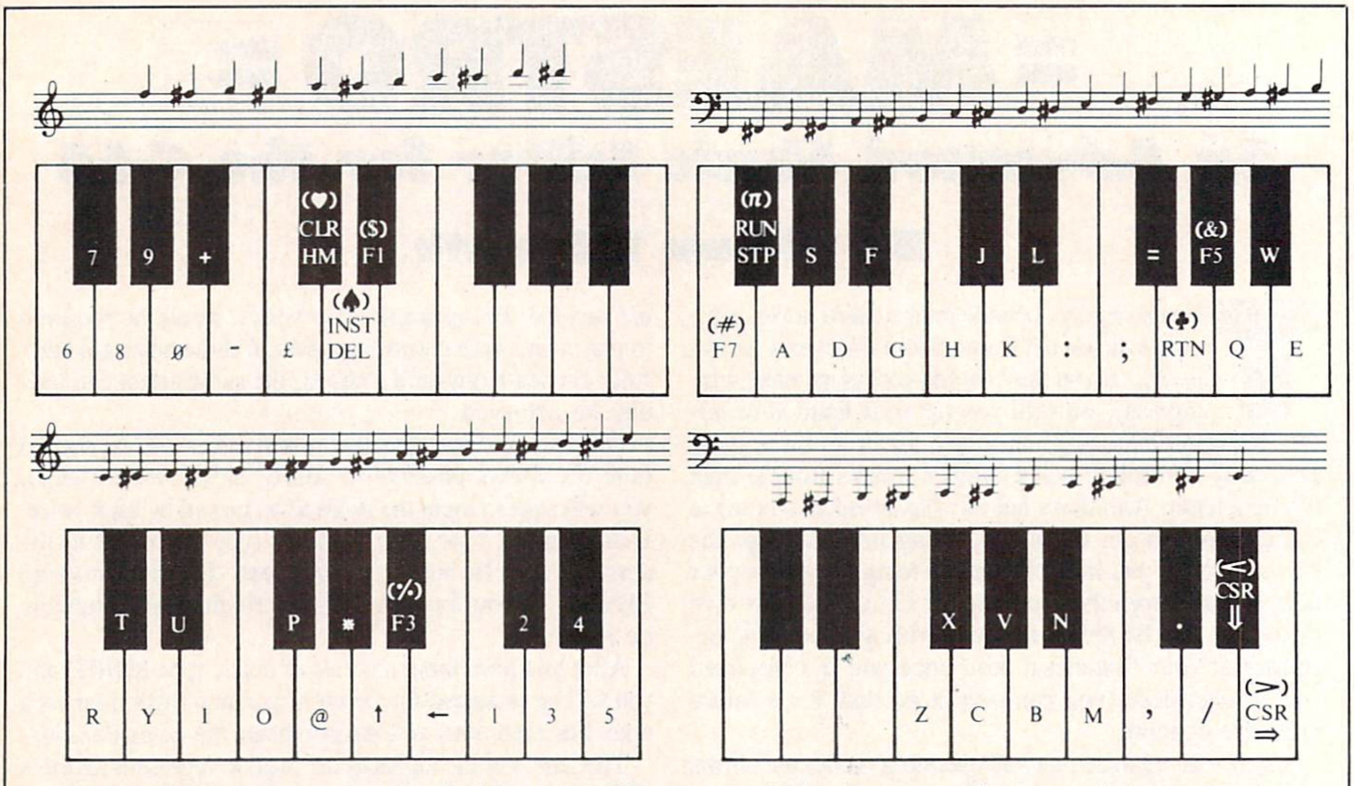
The letters of the alphabet are used to represent rhythms. The letter "A" represents the smallest unit of time in a piece of music, usually a sixteenth note. The letter "B" makes a note play twice as long as "A", "C" three times as long, and so on.

If "A" is used for a series of notes, they will slur, with no distinct separations in between. If this effect is not desired, use "B" for the shortest notes. Two typical rhythm assignments are shown here.

Typical Rhythm Assignments

RHYTHM	OPTION 1	OPTION 2
Sixteenth	A	B
Eighth	B	D
Dotted Eighth	C	F
Quarter	D	H
Dotted Quarter	F	L
Half	H	P
Dotted Half	L	X
Whole	P	...

If you accidentally type a wrong note or rhythm, simply press CURSOR LEFT while holding down the COMMODORE key. The note or rhythm will be cancelled, and you can then input the correct one. The CURSOR LEFT key cannot be used to correct notes after you have typed SHIFT and are prompted to input the rhythms. Furthermore, the CURSOR LEFT key cannot be used to correct the last rhythm entered. Once the last rhythm is typed, *Maestro* immediately begins to compile the music. But have no fear. From that point on, corrections can be made using the Music Editor (MODE 4).



Piano equivalents of notes played by C-64 keys under Maestro *control (also shown on staffs above each keyboard section).*

(Note: when MODE 2 is selected, any notes and rhythms previously entered will be erased unless you *immediately* exit the mode by typing SHIFT when prompted for the desired number of voices.)

MODE 3: LOOK AT MUSIC

This mode permits you to examine the notes and corresponding rhythms of any voice. Type SHIFT to return to the menu.

MODE 4: MUSIC EDITOR

The Music Editor permits four operations:

1. Add additional notes to any voice.
2. Change the notes and/or rhythm of an existing voice by retyping (from the beginning) the correct notes and/or rhythms.
3. Eliminate a voice.
4. Add an additional voice to a piece of music presently consisting of fewer than three voices.

For example, if you have composed a single-voice melody (using MODE 2) and wish to add a second voice for harmony, use the Music Editor to EDIT Voice 2. The Editor will allow you to input notes and rhythms for that new voice. Similarly, to eliminate an undesired voice, EDIT the voice: type "B" to edit from the beginning, and type SHIFT when prompted to enter notes and rhythms. Naturally, to eliminate all voices, it is easiest to start from scratch by using MODE 2.

MODE 5: SELECT INSTRUMENT

Select the desired instrument for each voice by typing a number key. 10 instrument sounds are available. For greater musical color, try using a different instrument for each voice.

MODE 6: LISTEN TO MUSIC

In this mode, select the desired tempo (0 being the fastest), and indicate which voice you wish to hear. In this way it is possible to listen to each voice separately. To hear all voices simultaneously, type "4".

To stop music that is being played, simply type SHIFT.

MODE 7: SAVE MUSIC ON DISK

Reserve a spare formatted disk for saving music. Use this mode to name and save music on that disk.

MODE 8: GET MUSIC FROM DISK

Load music you have saved on disks. Other sequential files may be confused with music files, so do not attempt to load them.

MODE 9: LOOK AT DISK INDEX

Examine the titles of music you have saved on disk.

MAESTRO SUMMARY

SHIFT	Return to menu Start playback (PLAY IT AGAIN!) Stop playback Ready to input rhythms
COMMODORE KEY	Stop playback (PLAY IT AGAIN!)

WHAT IS THIS GIRL DOING?

THE			GIRL				IS	
TH	AX	PAS	GG3	ER2	LL	PAS	IH	SS
29	15	04	34	52	45	04	12	55

	FEEDING							
PAS	FF	IY	PA3	DD2	IH	NG	PAS	BB1
04	40	19	02	33	12	44	04	28

BREAD						TO	
RR1	EH	EH	PA1	DD1	PAS	TT2	UW2
14	07	07	00	21	04	13	31

	A		DUCK			
PAS	EY	PAS	DD2	AA	KK2	PAS
04	20	04	33	24	41	04



You can now have your computer speak to you . . . with the inevitable "Kilobaud Monster." This remarkable breakthrough in engineering is now being offered to the general public by Computel, and is made possible at a fraction of the cost you'd expect to pay anywhere else. You will also learn how computers speak, what speech synthesis is, and the basics behind phonemics. It's a great educational tool for the individual and for the institution alike!

The Kilobaud Monster is a revolutionary item for the hobbyist, because now you can have many new resources available for experimentation. No longer is this field of science limited to the lab, and no longer is its cost limited to governmental research projects. You can make your computer speak with little effort, a lot easier than you think!

The quality of speech pronunciation is excellent. Anything imaginable may be spoken, from scientific thru slang words—we include detailed instructions showing you exactly how to do this. You can now have all the satisfaction of doing this yourself, at a fraction of the cost, and having something really useful to show for your efforts!

VISA

MasterCard

YOUR COMPUTER WILL TELL YOU! DON'T THEY ALL NOW "SPEAK?". . .

Devant si je Puis

KILOBAUD MONSTER?

ONLY $9.95 *regular price* *

Introductory package includes everything you need to get started!

** SAVE—Computel members only $ 7.95.*

- Able to work with any computer!
- Complete instructions included!
- User-programmable vocabulary of virtually unlimited size!
- Program it to say anything you want!
- Guaranteed to hold your interest and delight for hours!

The Kilobaud Monster is the basis of a series of projects that can make your computer talk, and is necessary for later projects. The Kilobaud Monster will set you up with everything you need to know in order to attach it to your computer and enable it to speak words. All additional information is included with this beginner's package. The built-in editor in the Wordmaker enables you to easily "fine tune" every word you create to make it sound exactly as you wish. Or you can set up a "word bank" with the Message Talker that includes all the phrases and statements that you'll ever need! The Artificial Intelligence accessory will permit your computer to produce totally random sentences, following all the rules of correct grammar within a pre-defined vocabulary-set, all day long!

If you're new to the computer field, or if you're an old-time "hacker," you'll want to subscribe to our publication "Computel," which covers everything that YOU want to see! Encompassing the subjects of Computers, Electronics, and Telecommunications, this publication gives you a "new meaning" of the free enterprise system and the monopoly of existing organizations. You'll need to become part of this international hobbyist phenomenon really soon. SAVE $2.00 on EACH ITEM you buy if you subscribe to Computel now! And this is IN ADDITION to the total package price for the Complete KILOBAUD MONSTER Collection—you'll SAVE OVER $18. Dealer and educational institution inquiries invited. Circle our Reader Service Number for additional information.

You may order these products from us by sending a check, money order, postage or cash in U.S. negotiable funds only, to the address indicated below. Or you may charge your order to your Visa or Master Card. Please include $2.00 shipping and handling on all orders. California residents please add sales tax. Allow three weeks for delivery. Add $5.00 for next-day *shipment* by U.P.S.; $15.00 for next-day *delivery* by U.S.P.S. We guarantee your satisfaction for 90 days with our products, and offer a money-back guarantee if not fully satisfied during that period.

COMPUTEL PUBLISHING SOCIETY
6354 VAN NUYS BLVD., SUITE 161 / VAN NUYS, CA 91401-2696

YES, I'm interested in making my computer talk. Please send me the following items:

- ☐ INTRODUCTORY PACKAGE .... $ 9.95
- ☐ WORDMAKER .... $ 19.95
- ☐ MESSAGE TALKER .... $ 14.95
- ☐ ARTIFICIAL INTELLIGENCE.......... $ 24.95
- ☐ COMPLETE COLLECTION SET (SAVE $10) $ 59.50
- ☐ One year subscription to Computel,. $ 14.00
- ☐ including all Kilobaud Monster items $ 65.50

PLEASE ADD $2.00 SHIPPING & HANDLING TO YOUR ORDER.

CREDIT CARD__________ EXP. DATE______

NAME__________

ADDRESS__________

CITY______ STATE____ ZIP_____

Your Computer__________

SEND ALL PAYMENT IN U.S. FUNDS DRAWN ON U.S. BANKS • SORRY, NO C.O.D.'S

Make checks payable to: COMPUTEL PUBLISHING SOCIETY
Post Office Drawer 7765 / VAN NUYS, CA 91409-7765

COMMODORE KEY/ CSR RT	Erase note or rhythm
SPACE	Rest (silence)

INSTRUMENTS:

0	Accordian	5	Drum
1	Organ	6	Guitar
2	Clarinet	7	Xylophone
3	Synthesizer	8	Banjo
4	Kazoo	9	Piano

HELPFUL HINTS

When inputting notes, keep in mind that *no SHIFTed characters are used to represent notes* (for obvious reasons!).

After using MODE 4 to add a voice, be sure to assign it an instrument using MODE 5.

Do not use rhythm "A" when the same note is to be sounded twice in a row.

If no sound is heard when playing notes in MODE 0, 1, or 2, try changing instruments.

EXAMPLE MUSIC

From the menu, select MODE 2 (COMPOSE A TUNE). Then type 2 to indicate that two voices will be used, and input the following notes:

```
●I↑1←I←315*5↑I↑1←I←31↑●●515↑1I@◠↑363←3@←
Y◠I@151↑1◠3←@←I1↑◠↑Y←1
```

(From Figure 1 or 2 we know that 0 = SPACE)

Now type SHIFT, and enter the following rhythms:

```
BBBBBBBBBDDDDBBBBBBBBBDDHBBBBBBBBBDDDFBBBBB
BBDDDFBBBDFBBBDFBBBDDH
```

In a moment, you will be prompted to indicate a tempo for the immediate playback of Voice 1. Type any number and playback will begin. (To halt playback type SHIFT).

Now you are prompted to input the notes and rhythms for Voice 2. As before, type the following:

```
DQ&Q:QRE:EYRQ&:Q:QRE:EYRQRQYQ;QK;DHGK♣E♣
:♣H:AGDHK;GKAGH:DH#/A♣;♣H
```

(Here again, ♣ =RTN, & = f5, and # = f7.)

Type SHIFT and input these rhythms:

```
DHDBBBBBBBBBDDDDBBBBBBBBBDDDDBBBBBBBBBDDDFB
BBBBBBBDDBBBBBDDBBBBBDDBBBBH
```

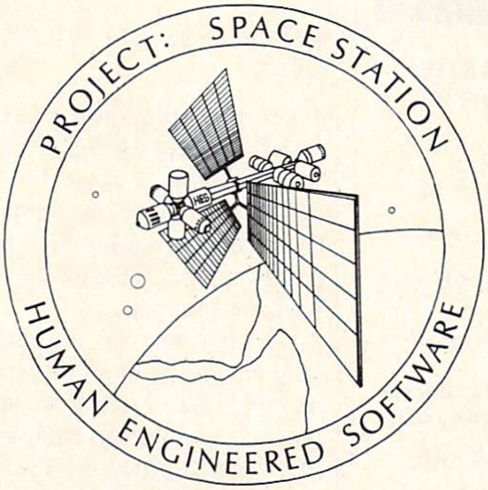
And that's all there is to it. You will be prompted to indicate a tempo for the immediate playback of Voice 2. Select any tempo, then type SHIFT to halt playback and move on.

Now, assign instruments using MODE 5. For this piece you might like 2-Clarinet for Voice 1 and 1-Organ for Voice 2.

Ready? You should see the menu once again. To hear all those notes and rhythms you have just provided, type 6 (LISTEN TO MUSIC), then 4 (both voices at once) and then whatever tempo you prefer (2 is good for this piece). Sit back and enjoy! □

SEE PROGRAM LISTING ON PAGE 134

PROJECT:
SPACE STATION™



THE ULTIMATE CONSTRUCTION SET.

Why hassle with breaking NASA's secret access codes, when you can design, launch, assemble and operate your own space station programs?

Have your friends over ... instead of the F.B.I.

WARNING: THIS IS NOT A TOY!

Apple II
Commodore 64

OTHER TITLES AVAILABLE FOR COMMODORE	
Cell Defence	14.95
Super Zaxxon	14.95
Graphic Basic	14.95
Paint Brush	14.95
Hesmon 64	24.95
OmniWriter	34.95

P. O. Box 631
S. San Francisco, CA 94083-0631

HesWare™

telephone: 415-871-0570
telex: 5101010530

Reader Service No. 236

Confused by Mail Order Ads???

Price is not the only concern. You should ask . . .

1. *Do you have the advertised product in stock?*
2. *If the product is defective who handles the warranty?*
3. *What is the total price delivered to my door?*

For the straight answers to these and all your questions, call us!

NEW C-128!!!
Call for pricing, availability & software.



Send Self Address Stamped Business Envelope for Complete Product List!!! C-64 Hardware still available, Call!

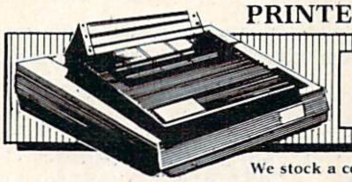
C-128 HARDWARE

1700 . . . . 128K Expander
1750 . . . . 384K Expander
1571 Drive . . . . 350K DS/DD
1670 Auto Modem . . 1200 Baud
1902 Color Monitor 13" RGB/Composite . . . 80/40 Column Display
Teknika MJ22 Same Specs as 1902 . . . . **$289**
1350 Mouse . . . . Eases Computer Input

C-128 SOFTWARE

JANE . . . . Integrated Software for use with 1350
Perfect Calc/Writer/Filer, Powerful CP/M Business Software

PRINTERS



STAR SG10
2K Buffer
NLQ Mode
CALL!

We stock a complete selection of ribbons

Powertype Daisywheel 18 CPS with Cardco B **$354**
DPS 1101 Daisywheel Direct Connect to C64 . . . . . **Call**
Okidata 182 . . . . **Call**
MW-350 Interface . . . . **Call**
Riteman C+ . . . . **Call**

MONITORS

Zenith ZVM Gr/Amb . **$89/99**
Teknika MJ10 . . . . . . **$189**

MODEMS

Commodore 1660 . . . . **$59**
Mighty Mo . . . . **Call**
Westridge 6420 . . . . . . **Call**
VIP Terminal . . . . **Call**
CompuServe Starter Kit & Vidtex . . . . Both for **39.95**

ACCESSORIES

Com Cool Drive Fan **$49**
Com Cool Plus w/ surge . . . **69**
Power Pak **$27**
Surge Protector 4-Plug . . . **25**

• *SPECIAL* • **Xidex Diskettes SS/DD $10.95** • *SPECIAL* •

BLACK NAUGAHYDE COVERS

C-128 or 1571 Drive . . . . ea. **$ 7**
C-64 or 1541 Drive . . . ea. **$5**
MSD Drives 1/2 . . . . **5/6**
Gemini 10X **9**
1702 Monitor **10**
1525/MPS 801 . . . . **7**
1526/MPS 802 . . . . **8**

DATABASE MANAGERS

The Consultant **$59**
Super Base 64 . . . . **49**
PFS:File **$59**
PFS:Report **49**

WORD PROCESSORS

Paper Clip/& speller **$55/69**
Fleet System 2 . . . . **Call**
Easy Script **$35**
Easy Spell **17**

MOST ORDERS SHIPPED WITHIN 48 HOURS! Advertised prices are for bank check or money order. VISA/MC orders add 3.5%. NO C.O.D ORDERS. Personal or company checks delay orders 21 days. All sales are final – defective merchandise exchanged for same product only. Call for return authorization number. All returned products are subject to a restocking fee unless they are defective and returned for exact exchange. In the continental U.S. add 3% for shipping ($2.50 min.) Please call for shipping on monitors. Ohio residents add 5.5% sales tax. Prices & availability subject to change.

HOURS: MONDAY-FRIDAY 11 AM to 7 PM • SATURDAY 10 AM to 5 PM

ORDER LINE: **800-638-2617**
INFORMATION & IN OHIO 216/758-0009

BIG BYTES

1301 BOARDMAN-POLAND ROAD POLAND, OHIO 44514

Reader Service No. 244

ERRATA

Line 685 of *Formatter* (August) was omitted during printing. It should read

```
685 GET D$: IF D$="" THEN
685
```

Users of some newer C-64's will find that *Gators N Snakes* (August) will occasionally cause the screen to flash, then blank out with unusual characters appearing. To correct the problem, turn off the machine and reload the program.

We apologize for any inconvenience caused by these problems. Remember that corrections to *Ahoy!* programs and articles are listed on the *Ahoy!* Bulletin Board (718-383-8909 – modem required) as soon as they are spotted.

SMALL THINGS CONSIDERED

Ahoy! is pleased to congratulate

Ari Everett
New York, NY

winner of a subscription to *Ahoy!* courtesy of New York's *Small Things Considered* radio show (heard weeknights 5-8 and Saturdays 6-8 on WNYC AM83).

AHOY!'S BBS

If your computer is equipped with a modem, you can call *Ahoy!*'s Bulletin Board System (718-383-8909) any hour of the day, any day of the week to exchange electronic mail with other Commodore users and download files featuring editorial calendars for upcoming issues, excerpts from future editions of *Scuttlebutt,* and more.

Set your modem for 300 baud, full duplex, no parity, one stop bit, 8-bit word length, and dial away!

TIPS AHOY!

We kept our part of the bargain—we came up with the title for *Ahoy!'s* programming tips column. And many of you readers have kept your part of the bargain, by supplying us with useful hints, tips, and programming routines for the Commodore computers—many more than we can fit into the limited space available in this issue. But *Tips Ahoy!* will be expanding in future issues, and we hope to receive input from a great many more of you. We'll pay top industry rates for accepted submissions. Send your best to *Tips Ahoy!*, Ion International Inc., 45 West 34th Street—Suite 407, New York, NY 10001.

LIFESAVERS TO SCOTCH TAPE TO YOUR COMMODORE

Many readers have called up or written regarding problems entering the BUG REPELLENT program. The most typical error is an:

```
?ILLEGAL QUANTITY ERROR
IN 5000
```

Until now there was no way to know where the bad data statement was, but with one simple command typed in immediate mode you can tell what line number the bad data element is in. Simply use the following line:

```
PRINT PEEK(63) + PEEK(64)*
256 [RETURN]
```

The computer will return the line containing the error.

If you have ever inadvertently typed NEW, the following line will restore your program:

```
POKE2050,1:SYS42291:POKE45
,PEEK(34):POKE46,PEEK(35):
CLR [RETURN]
```

Your program will be restored back into memory.
—H. Roth
Los Angeles, CA

HIDING YOUR PROGRAMS

If you wish to make a program "unlistable" enter the following line into your program:

```
1 REM (SHIFT L)
```

When someone types LIST the computer will show

```
1 REM

? SYNTAX ERROR
```

If people wise up to this trick, you may wish to be a bit more devious. Put the REM statement with the (SHIFT L) in the same line as an important statement, such as

```
100 V=53248:REM [SHIFT L]
```

If this line is deleted in order to view the program, the variable will not be defined and the program will not run.

To make your program unstoppable once running, use POKE 788,52. This will disable the RUN/STOP key.
—Craig Buchaman
Huntsville, AL

AUTO RUNNER

Instead of having to LOAD and RUN a program in separate steps, typing

```
LOAD"(FILENAME)",8,1(SHIFT
)(RUN/STOP)
```

will LOAD and RUN a program automatically.
—Barry Allyn
Arlington, VA

UNDIM YOUR DIMS

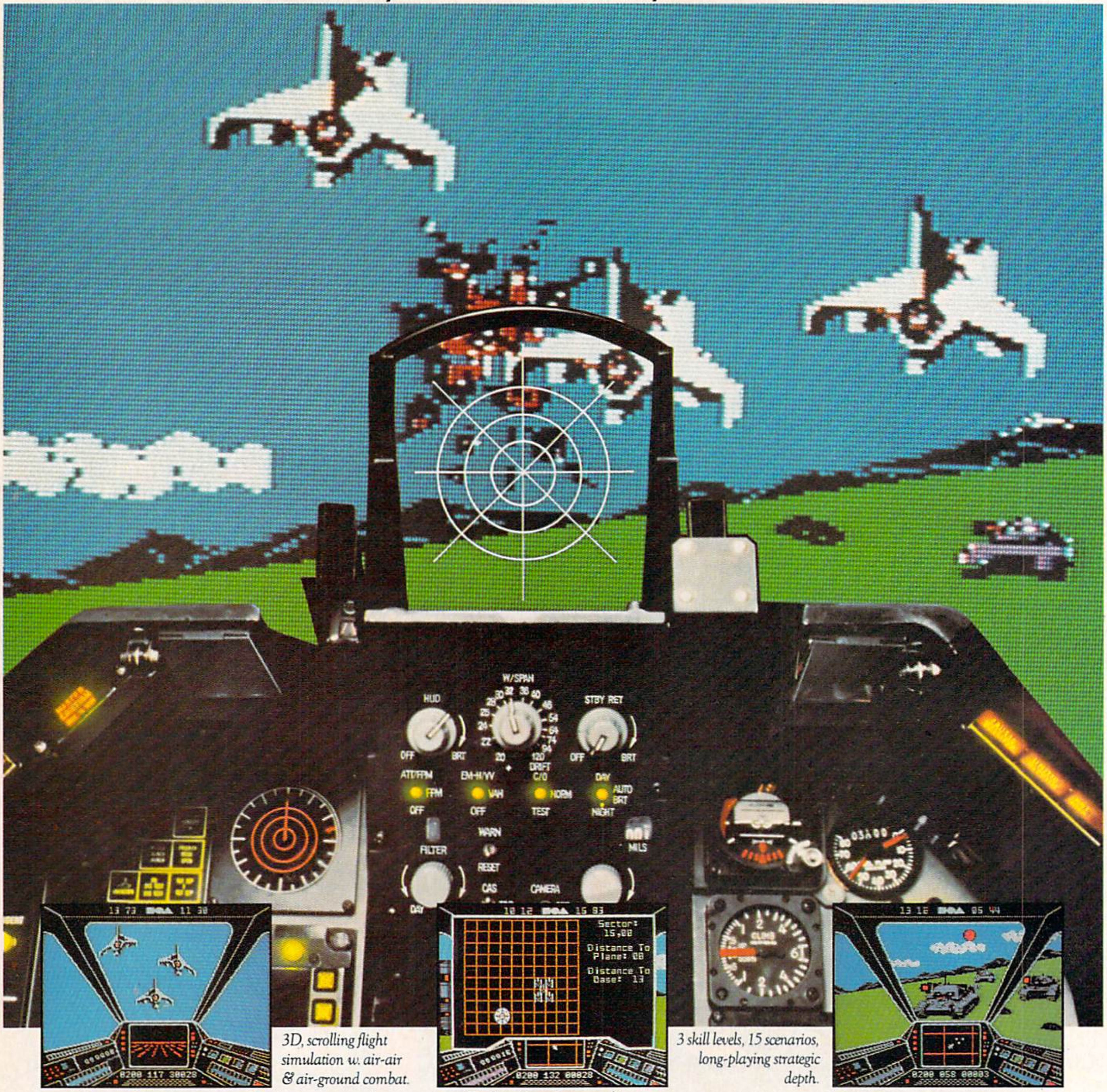
One limitation of BASIC on the C-64 is the inability to redimension an array. The following line will allow you to undimension an array so that you may redimension it.

```
POKE 49,PEEK (47):POKE 50,
PEEK (48)
```

—Jerry Jarvis
Spokane, WA


MASTERTR
ARCADE-STRATEGY-ACTION
Sure Chart Winners
VEGAS JACKPOT
One of the most faithful computer simulations ever produced of a fruit machine. It incorporates the hold, nudge, gamble and shuffle features found in the real thing.
SPACE WALK
A satellite retrieval game that includes 16 levels of "low pressure" death awaiting the unskilled astronaut.
BMX RACERS
A sure chart winner. You will need skill and fast reflexes to beat this one! You can ride on 5 different courses, each course ranges from 8 to 25 different scrolling screens.
CHILLER
Opened to Five Star Reviews in Europe recently this is surely one of the most challenging and unusual games available for the C64. Written by "Commodore Programmers of the Year" Richard and David Darling. The game has five different spectacular screens with hidden twists — if you get this far you have to retrace your steps through another five of increasing difficulty.
MIND CONTROL
Another unusual arcade — strategy game in which the players strive by weird and wonderful techniques to achieve a state of miniaturisation and hence enter Zyco's brain and the game continues in another dimension.
MAGIC CARPET
An imaginative three screen multi level arcade adventure that uses very high-resolution graphics and will test the abilities of the most experienced games players.
1985 THE DAY AFTER
A scrolling multi screen game that takes Commodore graphics to a 'new' state of the art level previously seen only in a handful of other games. Playability is a keyword on this 'five star game'!
KIKSTART Off-Road Simulator
A one or two player game with split screen scrolling graphics that realistically simulates off road riding on eight different and challenging courses.
More Adventures of BIG-MAC the Mad Maintenance Man
Multi screen arcade action with real twists needing a steady hand and a bold strategic game play. It will keep your interest alive for hours.
challenger
Neat variation on a city bomber theme using excellent scrolling graphic routines that is extremely addictive.
MAGRA and GAMES CREATOR $19.99
$9.99
SPECTACULAR
HIGH QUALITY SOFTWARE

Fly the unfriendly skies.



3D, scrolling flight simulation w. air-air & air-ground combat.

On-board computer, working gauges & 3 weapon systems.

3 skill levels, 15 scenarios, long-playing strategic depth.

You're in the cockpit of a dream machine—a bad dream for the poor sap whose tail you're waxing. You stick a silver bird up his exhaust and wince as he blossoms into fire. And then you rain down like pestilence upon the grunts in the tanks.

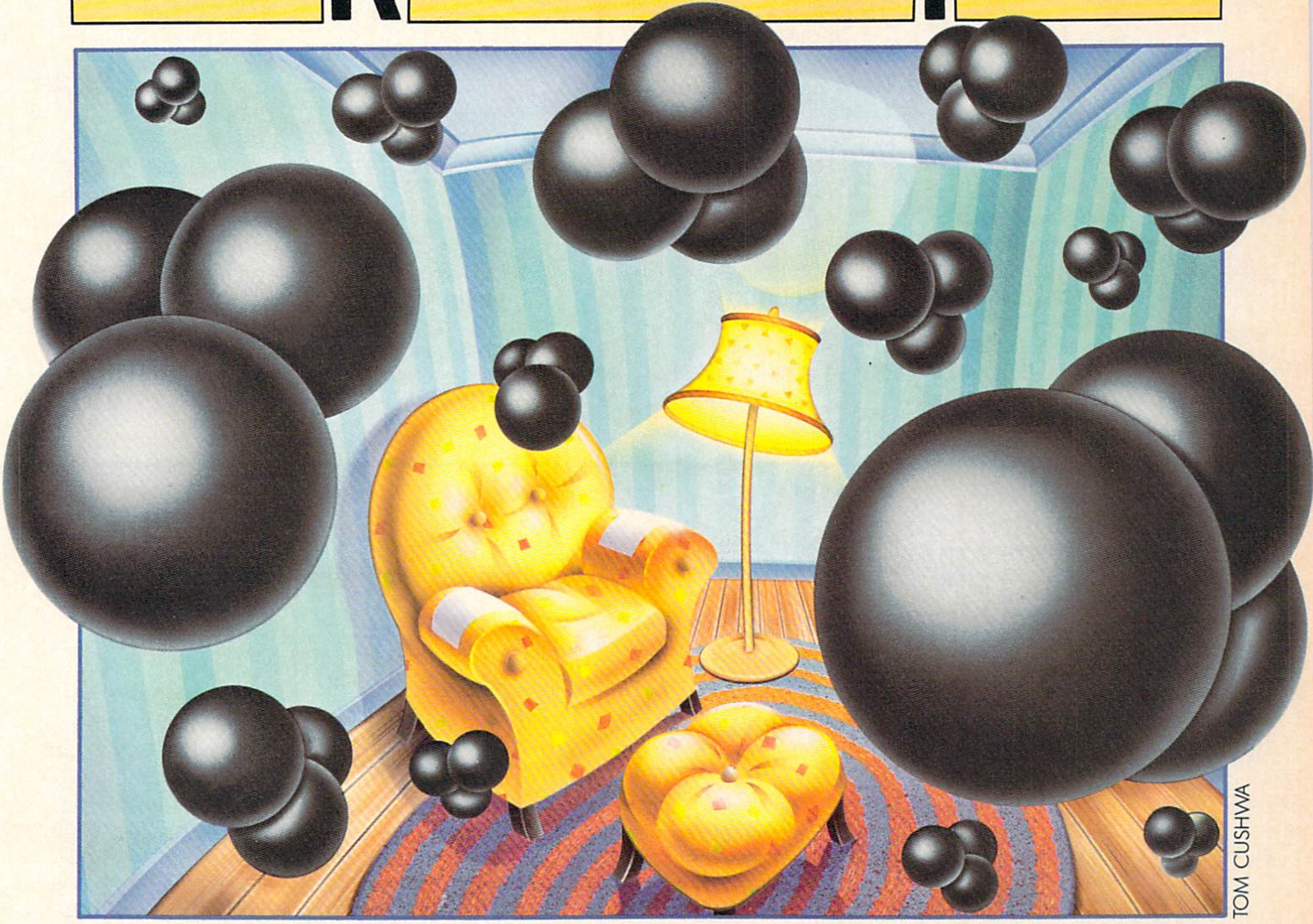
Skyfox. It's the fastest-selling EA game in history. It has the most awesome, high-speed animation you've ever seen on your computer. It's played to rave reviews in every magazine. And it's yours—at your retailer—or direct from EA at (800) 227-6703 with a 14-day satisfaction or money-back guarantee.

SKYFOX™

from ELECTRONIC ARTS™

ORDERING INFORMATION & MACHINE AVAILABILITY: Visit your retailer or call (800) 227-6703 (in CA call (800) 632-7979) for VISA or MasterCard orders. To purchase by mail, send check or money order to Electronic Arts, P.O. Box 306, Half Moon Bay, CA 94019. Add $3 for insured shipping & handling. Indicate machine verson desired. Allow 1-4 weeks for delivery. Apple II family available now at $39.95. C-64 available now at $34.95. For a copy of our complete catalogue and direct order form, send 50¢ and a stamped, self-addressed envelope to Electronic Arts, 2755 Campus Drive, San Mateo, CA 94403.

Reader Service No. 124



Infinitesimal Intrigue

By Dale Rupert

Computer Explorations of Sub-microscopic Activity

How often do you get the chance to watch molecules in motion, or to monitor short-lived radioactive isotopes? This month you can add those two activities to your list for the skeptic who asks, "What can you do with a personal computer, anyway?"

We will continue exploring ways to simulate the real world with the microcomputer. These investigations may not uncover any revolutionary, fundamental truths (although they may). Our goal is simply to gain some insight into the behavior of objects beyond the realm of our normal senses.

Suppose you could see the molecules of air in a room. Undoubtedly they would be scattered randomly and rather uniformly. Now imagine that as you watch, all the molecules gradually migrate to the left half of the room, leaving a vacuum in the right half.

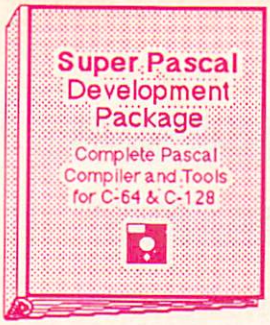
Is there any reason this couldn't happen? In fact, from a physical standpoint, it certainly *could* happen. Each molecule has a certain velocity. If the velocities of all molecules just happened to be in the same direction at the same time, theoretically all the air could travel to one side of the room.

From a statistical standpoint it's a different story. The

SUPER LANGUAGES



The most advanced **C** development package available for the C-64 or C-128 with very complete source editor; full K&R compiler (w/o bit fields); linker (binds up to 7 separate modules); and set of disk utilities. Very complete editor handles search/replace, 80 column display with horizontal scrolling and 41K source files. The I/O library supports standard functions like printf and fprintf. Free runtime package included. For C-64/C-128 with 1541/1571 drive. Includes system diskette and user's handbook. **$79.95**



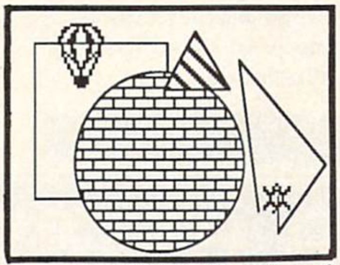
Not just a compiler, but a complete development system. Rivals Turbo Pascal© in both speed and features. Produces fast 6510 machine code. Includes advanced source file editor; full Jensen & Wirth compiler with system programming extensions, new high speed DOS (3 times faster); builtin assembler for specialized requirements. Overlays, 11-digit arithmetic, debugging tools, graphics routines, much more. Free runtime package. Includes system diskette and complete user's handbook. **$59.95**

BASIC-64 full compiler

```
ADVANCED DEVELOPMENT PACKAGE
A = CODE-GENERATOR:        P-CODE
B = LOAD SYMBOL-TABLE:     OFF
C = SAVE SYMBOL-TABLE:     OFF
D = LINE-ADDRESS-TABLE:    OFF
E = MEMORY-TOP:            65536
F = CODE-START:            7557
G = RUNTIME-MODULE:        ON
H = EXTENSION:             SIMON'S BASIC
I = TOKEN-BYTES:           2
J = ELSE-CODE:             100  71
K = ERROR-LINE:            0
L = OVERLAY:               OFF
M = DISK-COMMAND:
```

The most advanced **BASIC** compiler available for the C-64. Our bestselling software product. Compiles to superfast 6510 machine code or very compact speedcode. You can even mix the two in one program. Compiles the complete BASIC language. Flexible memory management and overlay options make it perfect for all program development needs. **BASIC 64** increases the speed of your programs from 3 to 20 times. Free runtime package. Includes system diskette and user's handbook. **$39.95**

VIDEO BASIC development



The most advanced graphics development package available for the C-64. Adds dozens of powerful commands to standard BASIC so that you can use the hidden graphics and sound capabilities. Commands for hires, multicolor, sprite and turtle graphics, simple and complex music and sound, hardcopy to most printers, memory management, more. Used by professional programmers for commerical software development. Free runtime package. Includes system diskette and user's handbook. **$39.95**

FORTH Language

```
SCR # 20
 0 P ( RANDOM NUMBER TESTER ?RND )
 1 P FORTH DEFINITIONS DECIMAL
 2 P : ?RND
 3 P        ( INITIALIZE FIRST SCREEN)
 4 P        1024 1000 ASCII 0 FILL
 5 P        BEGIN
 6 P           1000 RND  ( RANDOM 0.999)
 7 P           40 /MOD   ( COLUMN, LINE)
 8 P          SWAP      ( EXCHANGE)
 9 P           2DUP S@   ( CHARACTER)
10 P           1+ -ROT   ( ADD 1)
11 P           S!        ( SAVE)
12 P        ?TERMINAL UNTIL
13 P ;
```

Our **FORTH** language is based on the Forth 79 standard, but also includes much of the 83 level to give you 3 times vocabulary of fig-Forth. Includes full-screen editor, complete Forth-style assembler, set of programming tools and numerous sample programs to get you deeply involved in the FORTH language. Our enhanced vocabulary supports both hires and lores graphics and the sound synthesizer. Includes system diskette with sample programs and user's handbook. **$39.95**

Other software also available!
Call now for free catalog and the name of your nearest dealer. Phone: **616/241-5510.**

Abacus Software

P.O. Box 7211 Grand Rapids, MI 49510 616/241-5510

For fast service call **616/241-5510.** For postage and handling, include $4.00 per order. Foreign orders include $8.00 per item. Money orders and checks in U.S. dollars only. Mastercard, Visa and Amex accepted.

Dealer Inquiries Welcome
More than 1200 dealers nationwide

Reader Service No. 213

PROGRAMMERS!

Ahoy! wishes to continue publishing the best Commodore-compatible programs available in magazines today! We're looking for games, utilities, educational, music, and graphics programs, and programs that completely defy description. If you have an original C-64, C-128, or VIC 20 program, in BASIC or machine language, we'd like to see it. Payment will be made upon acceptance, at competitive industry rates. You will also receive royalties based on the sale of our program disks.

Send a copy of your program on disk or tape, accompanied by a printout and documentation, to *Ahoy!*, 45 West 34th Street—Suite 407, New York, NY 10001. Include a self-addressed mailer with sufficient return postage affixed. (Program printouts not accompanied by a disk or tape will be returned unread; programs not accompanied by sufficient return postage will not be returned.)



"Don't be silly, Joyce...what could my new computer modem have to do with this $120 phone bill?"

Reader Service No. 245

likelihood of all air molecules in a room spontaneously having velocities in the same direction at the same time is negligibly small. The situation (on a smaller scale) is analogous to flipping a million coins and having them all come up tails. Possible? Yes. Probable? Don't bet on it.

TOWARD INCREASING DISORDER

If you watched a movie of thousands of identical molecules moving around in a container, would you be able to tell if the movie was going forward or backward? If the molecules were uniformly distributed and remained so throughout the movie, you certainly could not tell.

(You might prefer a simpler and more familiar example if you find it difficult to picture molecules moving around. Consider the wind-filled box containing Ping-Pong balls that is used for drawing lottery numbers or for choosing bingo values. The balls continuously float around, bouncing off the sides of the container and each other. This is not really an isolated system because of the external fan, but it gives you the idea of the random motion with which we will be dealing.)

Suppose that the molecules gradually began congregating in the left half of the container, until the final scene showed no molecules at all in the right half. Assuming the molecules have no attraction for each other, it doesn't require a course in statistical mechanics to realize that the movie was probably running backward.

We would conclude that the film started with all molecules on the left and that they migrated in all directions until they "filled" the container. The concept that things in the universe tend to become more disorganized or more randomized is perhaps common sense. The physicist says in the Second Law of Thermodynamics that the natural tendency is for the entropy of a system to increase. This is another way of expressing our notion that the amount of disorganization of the molecules in the container should be increasing. If we see the molecules organize themselves in one half of a container, it seems somewhat mysterious, or downright unnatural!

The program *Molecular Motion* on page 120 simulates the random movement of molecules within a container. (The program is only for the Commodore 64.) The table on this page lists the variables and their uses. The molecules all begin on the left side of the screen. A random number is chosen to decide if the next molecule to trade sides comes from the left or from the right side of the screen.

The probability that a molecule moves to the other side of the screen depends upon how many molecules are currently on its side of the screen. If all molecules are on the left, then obviously the next movement must be from the left side to the right side. If the molecules are evenly distributed, the next molecule to move to the other side is just as likely to be on the right side as on the left.

The relative number on the left (RL) is calculated in line 80 from the total number on the left (NL) divided by the total number (TTL) of molecules. Since all molecules start on the left side, this ratio is initially one. Line 90 picks a random number between zero and one. If the random number is less than the relative number on the left side, it is assumed that one of the molecules migrates from the left to the right.

As more molecules move to the right, the relative number on the left decreases. Consequently it becomes less probable that a molecule will move from left to right. For example, when half are on each side, the relative number on the left is one half. If the random number is less than one half, one moves from left to right. It is just as likely in this case that the random number will be greater than one half, meaning that one moves instead from right to left.

After the program has run for awhile, you should notice that the numbers on the left and right sides are nearly equal. They vary some as would be expected. Hopefully

VARIABLES FOR *MOLECULAR ACTION*

A$ keyboard input	**MR** memory to POKE on right
KK direction: 1=normal; 0=reversed	**NL** number on left
K1 start of screen memory	**PL** POKE character for left
K2 width of screen	**PN** random number
K3 blank character	**PR** POKE character for right
K4 molecule character	**RL** relative number on left
L$ format character	**T** time counter
L1$ cursor left	**TTL** total # of molecules
ML memory to POKE on left	**X** screen column (0-39)
	Y screen row (0-24)

you agree that it is most unlikely for the original distribution to occur again with all molecules on the left. Chances are equally remote that all of the molecules would ever end up on the right side of the screen at once.

THE ARROW OF TIME

Robert Eisberg describes this problem as a calculator simulation in his book *Applied Mathematical Physics with Programmable Pocket Calculators* (McGraw-Hill, 1976). He refers to "the arrow of time" as the natural direction for the system to progress.

We can easily simulate reversing the arrow of time. That way we will be able to make our "computer movie" run backward. Normally K3 represents the empty space character (ASCII value 32), and K4 represents the molecule (ASCII value 81). If any key is pressed while the program is running, the values of K3 and K4 are swapped in line 270.

Now when the test in line 110 says to move a molecule from left to right, the opposite will occur. A molecule will be put on the left and a blank will be put on the right. Now it is more likely for a molecule to go from right to left whenever there are already more molecules on the left. Let the program run for a while with the arrow of time reversed. All of the molecules regathering on one side of the screen seems unnatural. Pressing another key reswaps the values of K3 and K4 and sets the arrow of time straight again.

The numbers at the top of the screen tell the numbers of molecules on each side and the number of time intervals that have elapsed since the last key was pressed. Time is displayed as a negative value when the arrow of time has been reversed.

This concept of the increasing entropy or disorder of an isolated system relates only to systems containing reasonably large numbers of objects. If there were only two molecules in the container, it wouldn't seem unusual for both of them to be on the same side at one time. It would be impossible to know whether the movie is going forward or backward if it showed just two molecules. With 480 molecules, it should be readily apparent when things are behaving naturally and when they aren't.

RADIOISOTOPE ACTIVITY

Radioactive nuclei decay at random. A nucleus of Uranium 234 (U-234) splits into a Helium nucleus and a nucleus of Thorium 230. U-234 has a half-life of a quarter of a million years. This means that after that amount of time, half of the nuclei in the original sample of U-234 will have decayed.

Just as with the probabilities of coin flipping, you can't predict what any one nucleus will do at any given time. You can only come up with averages for many nuclei or many time intervals. A particular U-234 nucleus might decay today or it might wait several million years before decaying.

Thorium 230 is called a "daughter product" of U-234. Thorium 230 itself is radioactive. It breaks down into other products which eventually decay to become Lead 206 nuclei.

The program *Radioactive Decay* on page 120 simulates the decay of a sample of a radioactive substance. (This program is written for the Commodore 64. The first part of this program also runs on the VIC 20 if lines 240 and 250 are deleted. The Plot Routine works only on the Commodore 64.) The initial substance decays to one radioactive daughter product which itself decays to a stable second daughter product.

Common sense tells us that as the amount of the original substance decreases, the amount of the first daughter product should increase. Furthermore the amount of the stable second product should gradually increase as some of the first product decays. Eventually only the stable nucleus, the second daughter product, remains.

The relative amounts of each substance at any time depend upon the relative half-lives of each substance. If the initial substance decays rapidly and the first daughter product decays much more slowly, we would expect the amount of the first daughter product to increase rapidly. Its concentration should remain fairly high for quite a while as it slowly decays into the second daughter product.

The first time you run *Radioactive Decay,* line 170 calls the subroutine at line 400 to print the quantities of each type of nucleus. You may add an appropriate OPEN statement and change lines 400 and 420 to PRINT# state-

ments if you want the output to go to your printer instead of to the screen.

If you are using the Commodore 64, you may change line 170 to GOSUB 500. This calls the Plot Routine which creates a graph of the quantities of the three substances as a function of time. Pressing any key after the first screen is completed will continue the graph onto the next screen.

This program is easy to modify. First of all, the decay constants in lines 110 through 130 may be changed. A value of one means that the substance decays instantly. A value of zero means that the substance is stable. The closer the decay constant is to one, the shorter its half-life is. See how changing the decay constants changes the relative numbers of each type of isotope.

For the technically oriented readers, the relations between the decay constant (DC) and the half-life (t) are as follows:

$$DC = 1 - EXP(-0.693/t)$$
$$t = -0.693/LOG(1 - DC)$$

The number −0.693 is the natural log of one half.

Initially there are 100 type 0 nuclei. The computer selects a random number for each nucleus. If that number is less than the decay constant for a given nucleus, then that nucleus decays. A type 0 nucleus becomes a type 1 nucleus when it decays. A type 1 nucleus decays into a type 2 nucleus. Since the type 2 nucleus has a decay constant of zero, it will never decay.

The array NUC( ) keeps track of each nucleus' type. For example, if the tenth nucleus has decayed twice, it is a type 2 nucleus, and NUC(10) equals two. The array M( ) keeps track of the total numbers of each type of nucleus. Initially M(0) is 100 while M(1) are each zero, since all 100 nuclei are type 0. When a type 1 nucleus decays, there is one less type 1 and one more type 2. Line 210 does the accounting.

It is not difficult to change the initial number of nuclei or the number of different daughter products. Line 10 defines the total number and line 30 gives the number of daughter products. If you change the number of daughters, you should change the numbers of decay products beginning at line 110. There should be a decay constant for each type of nucleus.

John R. Merrill provides a more technical discussion of radioactivity simulations and many other topics in his extensive book *Using Computers in Physics* (Houghton Mifflin, 1976). You could certainly learn a lot of physics by running the programs in his outstanding book.

As usual, the programs this month are merely a starting place for your investigations. Simulations on your microcomputer provide a window into the workings of the universe. Enjoy your explorations and your discoveries. □

SEE PROGRAM LISTING ON PAGE 120

COPY PROTECTION: THE GUARDED SECRETS

Including *Disk Error Checker,* a Utility for the C-64

By George Trepal

It seems that every other ad in computer magazines is for programs that copy other programs (but not themselves). What's going on? Why should a program be hard to copy, anyway? Shouldn't a simple LOAD then SAVE to another disk make a copy?

Imagine being given the job of designing a book that anyone could read but no one could copy. Making software copyproof is a lot like this. The computer must read the information from the disk in order to run the program, yet the information must somehow be protected from being copied. When the program is in the computer it can alter the computer's SAVE and BREAK instructions and protect itself. When it's on the disk it has no such protection and can be copied a block at a time. Yet there are programs that can't be copied.

To understand what's happening we have to start with the disk drive. It's not simply a box. Inside is a full computer with ROM and RAM and a dull little filing program that runs constantly.

In addition to the computer circuitry is the disk turning motor and a system to move the magnetic head over the disk. The head is responsible for reading or writing information to the disk. The head moves in steps across the disk with each step representing a track. The steps are referenced to solid objects called stops. When you format a disk the drive starts the process with internal banging. Each bang is the head whacking against a stop to make sure it's gone as far as it can. When it is satisfied about its position it starts laying down tracks.

If you're thinking that this is a pretty sloppy way to run things you're right. If the head can't position itself over the tracks it's said to be out of alignment. Since the alignment for each drive is slightly different, a disk that works on one drive may not be readable by another.

If you'll look at a disk you'll see a little round hole near the hub. Some drives have a light beam-photocell combination to look for the hole. When the hole shows up, the computer has a reference point to work from. This is called hard sectoring a disk. Commodore computers use soft sectoring, which means the hole isn't used and the drive has no idea where it is on the disk. It gets around this problem by putting a marker called a sync character at the start of things. Each track is divided into sectors with the outer tracks having more sectors than the inner tracks. The sectors of a track take up equal room, and a space between the last and first sectors is provided as a safety margin. If the space weren't there and the drive's timing were off, the last sector could overwrite the first sector.

The computer and drive are both dedicated to 100.000% accuracy. A program that takes up 100 blocks on the disk represents about 25,000 bits of program information and 9,000 bits of error detection information. If even one is wrong, the program won't run right!

One way to detect errors is to add the values of all the bytes stored in a sector and store this checksum separately. When the information is read off the sector the addition is done again and compared with the checksum. If the two don't match, the information is read again, since an error has been detected. After a number of tries and failures, an error is declared and the red drive light starts to flash.

Each sector has the disk ID written into it. If the ID doesn't match the ID the computer found when it checked the directory, an error is declared.

Synch characters tell the drive where it is on the disk. If a synch character is missing you'll get an error.

Virtually all copy protection takes advantage of the fact that the drive and computer hate anything abnormal. Normal copy programs won't copy errors because—well, because they're errors and are therefore WRONG!

This is easier to understand if we humanize the machines. The scene: a program has been loaded into the computer and it has started to run.

Computer: Hmm! The first thing the program tells me to do is check to see if this disk is original or a copy. If it's a copy, I won't run the program. Hey! Drive, get me the data in sector one of track one.

Drive: OK, sector one of track one. Coming right up.

Unbeknownst to Drive, the entire track is empty and has been reformatted, thus providing it with an ID different from the rest of the disk.

Drive: Here it is. Better check to see if the ID matches. Oh, oh! it doesn't. My head is probably put out of alignment, so I'll whack it against the stop to align it. BANG! Nope, the error is still there so I'll whack my head again to align it. BANG!...

Drive whacks his head for some time. You hear it as about a second of machine gun fire. Eventually Drive gives up and an error is declared.

Drive: Yo, Computer! I can't give you the information. I tried hard to get it, but the ID doesn't match.

Computer: The program says that if Drive tells me there's an ID error, the disk is the original and it's okay to run

the program.

The thing that confuses some people is that the error is not part of the program. If the error were part of the program, the program couldn't be loaded, since it would have an error in it. The error is always confined to a portion of the disk not used to hold program information.

There are two ways to read errors and both are (poorly) covered in the Commodore drive instruction manual. One way is the block read just described. It involves a great deal of head banging, which can indeed cause head misalignment. All programs copy protected with block reads work to send your drive to the repair shop. The other method is called memory reading. A memory read actually looks inside the drive's buffers to extract information. There is no banging. The program that accompanies this article uses memory reads to check a disk for errors.

Simple errors were adequate protection two years ago, but now there are dozens of copy programs on the market that copy them. A war is on between those who want to protect software and those who want to copy it. Needless to say, almost no one wants to tell how they protect their software. The following information is probably right, but as I've said, people are reluctant to talk.

A disk normally has 35 tracks on it, but the head can be told to make extra tracks. Track 36 is the easiest to work with and track 40 is the hardest. Since a normal disk copy program only copies 35 tracks, vital program information stored on track 36 is lost. The error checking program I've included will check to track 40. If the tracks don't exist, the program will give error #21. If the tracks beyond 35 get a format command that tells the computer to lay down more blocks than there is room, the last blocks overwrite the first. This sort of "dirty format" will hang up my reader (and most copy programs).

There is room between the tracks for information, and this half-track space can be used to store critical data. I think true half-tracking is impossible with the 1541 drive since the head alignment isn't good enough for it. The half-traced track tends to destroy one of the tracks it's between. But that's taken into account. When a half-tracked program is copied, if the head alignment is very good the half-track is missed entirely. If it's normal, the head wiggles and picks up some information from the normal track and some from the half track, not getting clear information from either.

Similar to half-tracking is renumbering tracks, two tracks 34's for example. This can be terribly worrisome to copy programs.

Remember that the sync character lets the drive know where the information on the disk is. If a track sync character is removed, a normal copy program can never find where it is and hangs up. Of course, the track has no information on it.

Extra sectors can be put after the last sector of a track and information can be hidden in them.

The spacing of bits stored in the outer tracks differs from that of the inner tracks. By altering the bit density of a track it can be made unreadable.

There are a few other tricks (including the one my company uses) that I've promised not to write about.

A technique that has been proposed is to write a sector with a magnetic strength halfway between a 1 and a 0. When the computer reads this track it will read garbage which will vary with drive vibration, disk flexing, heat, and other factors. The sector will be read several times and if the result is different each time, the disk is assumed to be the original. If the disk were copied, the garbage read from the original would always be the same.

Another technique which is almost impossible to use with soft sectored disks is laser protection. A tiny hole representing one bit of a byte is burned in the disk. The byte is written with the last bit being 1. If it is read back as being 0 (i.e., hole) the disk is an original. This is really a technique for hard sectored disks.

There are two general classes of copy programs sold. The first sort reads information from a disk a block at a time and makes a copy of it. The second sort is called a nibble copier and is the meaner of the two. The destination disk is put in the drive and the drive rotation is timed. The source disk is then put in and a whole track (rather than block at a time) is read and stored to be fed to the destination disk later. Since the nibble copier doesn't have to worry about such things as synch characters, it's harder to protect against. Interestingly enough, most copiers tend to randomly introduce errors to the destination disk. Since most disks that are copy protected have lots of extra room on them, the introduced errors hardly ever hit a vital spot.

Why should errors be created? For one thing, the programs bypass the normal error traps to be able to write errors. Also, they bypass (usually) vital routines to gain speed. A normal copier written in BASIC takes about 25 minutes to copy a disk. By using machine language and bypassing things, a nibble copy of a disk (including half-tracks) takes about 8 minutes. Fast copy routines can do the job in about 3 minutes. Spoolers can do the job in about half a minute!

Spooling means to get peripheral devices talking to each other. A copy spooler uses two disk drives and has them sending information directly to each other. Once the process is underway, the computer can be turned off. Non-spooling copiers have to read the information into the computer, then out onto the destination disk.

Now to my error checking program. It only finds errors rather than writes them. What use could this be in making copies? Answer: Well, uh.... You may get a certain thrill in using it to find how commercial disks are protected. You can also use it to see if a disk, copy or not, has errors. If it is a copy, you can see if they are in the right places. You can use it to check formatted disks for bad spots. If you have critical information to save, it pays to check the disk first. The program doesn't hurt any information on the disk.

Copyright law allows you to make a copy of your copy protected disks for archival purposes.

You may want to format a disk and see how many errors you can create with a magnet. Be careful not to get it near the middle track since you may make the whole disk unreadable. You can also put blank formatted disks in different places to see if they are safe. Does it hurt to leave a disk on top of the monitor? How about under a ringing telephone? Are errors caused by freezing? What happens if a disk gets too hot? ☐ ***SEE PROGRAM LISTING ON PAGE 134***

WHITE HOUSE COMPUTER

P.O. BOX 4025, WILLIAMSPORT, PA 17701

In PA: 1-717-322-7700
1-800-351-3442
HOURS: Monday thru Friday 9:00 am till 6:00 pm
VISA and MC Accepted 4%

★ PRINTERS ★

C-64
MPS 801 .... 179.95
802 .... 208.95
803 .... 175.95
MCS 801 .... 314.95
DPS 1101 .... 314.95

CITIZEN
MSP-10 325
MSP-15 479
MSP-20 469
MSP-25 629

EPSON
RX80 219
LX80 249
FX80 369
JX80 529
RX100 369
FX100 550

C. ITOH
8510 A 285
8510 BC1 329
8510 BC2 389
8510 BP1 339
8510 SP 389
8510 SR 429
8510 SCP 459
8510 SCR 477
1550 BCD 485
1550 P 445
A10-20P 465
F10-40P/S 875
F10-55 P/S 1059
F10 Tractor 169
F10 Sheet Feeder 349

MANNSEMAN TALLY
Spirit 80 249
MT 160L 495
MT 180L 595

STAR MICRONICS
SG10 214
SG15 379
SD10 345
SD15 450
SR10 485
SR15 585
Powertype 309

CARDCO
LQ 1 365
★LQ 3 209

PANASONIC
1090 187
1091 237
1092 389
1093 585
3151 455

LEGEND
880 209
1080 229
1380 269
1385 305

OKIDATA
182 .... 219.95
84 .... 640.95
92 .... 349.95
93 .... 560.95
Imagewriter 425.95

DISK DRIVES
C-64
1541 .... 185.95
1571 .... 249.95
1572 .... 375.95
MSD Single 229.00
MSD Dual 459.00
Indus GT 239.00
Cardco SCD-1 .... 199.00

INTERFACES
Tymac Connection/2K 62.95
Cardco +G 59.95
Cardco B 37.95
OR-D GPC w/2K 55.95
Grappler CD 89.95

PRINTER PAPER
Lazor 1000 shts 15.95
Lazor 2500 shts 29.95

DISKETTES ★
SKC
SS/SD 10.50
SS/DD 13.50
DS/DD 15.50
MAWELL
MD-1 .... 15.95
MD-2 .... 21.95
ELEPHANT
SS/SD 13.50
SS/DD 15.50
DS/DD 19.50

INNOVATIVE CONCEPTS
Flip-N-File
10 .... 3.50
15 .... 8.50
25 .... 18.50
w/lock
50 .... 17.50
50 .... 23.50
ROM Holder 17.50

KOALA
Pad Disk 39.95
Pad ROM 45.95

MONITORS
AMDEK
300G 119
300A 129
310A 145
Color 300 235
Color 500 345
Color 600 399
Color 700 469
Color 710 539

TEKNIKA
MJ-10 199
MJ-22 285

XTRON
Comcolor I 189

ZENITH
122 .... 85.95
123 .... 75.95

BMC
Amber 82.95
Green 79.95

COMMODORE
C-1702 .... 199.95
C-1802 .... 204.95
C-1902 RGB 259.95
C-1901 Monchrome 129.95

Dealers Inquiries Invited ★



SOFTWARE ★

BATTERIES INCLUDED
Paper Clipw/Spell Pk 69.95
Paper Clip 59.00
Home Pack 32.95
Bus Card II 124.95
B180 Board 99.95

BRODERBUND
Print Shop 29.95
Graphics Library 18.95
B.S. Writer 32.95

EPYX
FAST LOAD 23.95

VIDEO CASSETTE RECORDER ★★ $275.95

Symphonic VHS/Remote
- 105 Channel Cable Ready
- 14 Day / 1 Program Timer
- 12 Pre-set Channels
- Auto Rewind

Plus much, much more!

CHECK OUR PRICES!

COMPUTERS
C-64 .... 149.95
C-128 .... 275.95

PRINTER RIBBONS

DUST COVERS

MODEMS
C-1600 .... 44.95
C-1650 .... 54.95
C-1660 .... 54.95
C-1670 .... 185.95
300/1200 Band

Cardco
MOD-1 .... 59.95
C-64
MPP-1064 .... 65.95
Tele Learning 49.95

Mitey Mo 59.95
Westridge 69.95
Compuserve Starter Kit 21.95

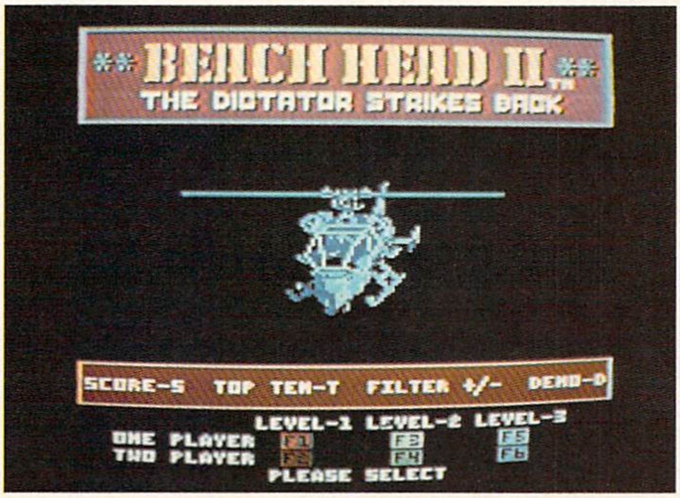
WE CARRY A FULL LINE OF:
Batteries Inc., Broderbund, Contineal, Hess, Infocom, Innovative Concepts, Koala, LJK, Microbits, Micropose, Microtek, O.S.S., Professional S.W. Scarborough, S.S.I., Synapse, Time Works, Wico.

POLICY: No deposit on COD orders. FREE freight on all prepaid cash orders over $300 in the continental USA. APO and FPO add $5 per hundred. For priority Mail add $8 per hundred. PA Residents add 6% sales tax. Defective products must have Prior RA number. Schools net 15.

Reader Service No. 220

BEACH-HEAD II

Access Software
Commodore 64
Disk; $39.95



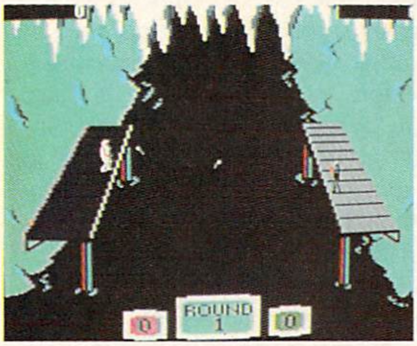
As any fan of movie serials knows, defeating a villain is one thing, keeping him down is quite another. *Beach-Head II* proves the same is true in the world of computer entertainment.

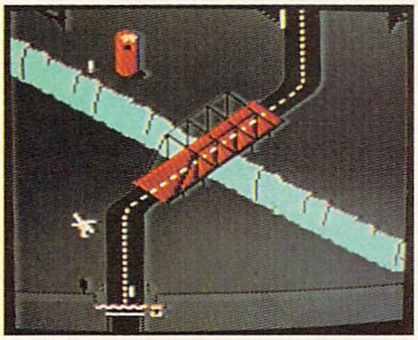
The Dictator Strikes Back is the ominous subtitle of this action strategy simulation of small-unit combat at the end of World War II. The implacable enemy, whom many computerists defeated in 1984's *Beach-Head,* has survived the destruction of his fortress. Now "The Dragon" is spoiling for revenge.

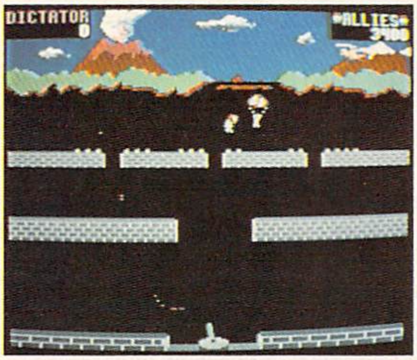
One major difference between *Beach-Head II* and its predecessor is that it can be played head-to-head as well as solitaire. In the solo mode, the gamer can control either the Allies or the Dictator in the program's four linked scenarios.

The only exception, a slap in the face to logic, is "Escape." The human player always controls the rescue helicopters against computer-directed ground forces. Even if the computerist takes the Dictator's side in the other scenarios, he or she must control the chopper in "Escape." There is no sensible explanation for the Dictator's sudden desire to ferry hostages to safety. The instruction booklet doesn't even try to offer a rationale. Possibly "The Dragon" intends to ransom his prisoners back to the Allies at a later date.

No one can accuse designers Roger and Bruce Carver of leading with anything less than their best. "Attack," the initial assault on the Dictator's sanctuary, is the most satisfying part of the game from several standpoints. It depicts an intriguing situation, has superlative audiovisual effects, and features the most original play-mechanic in the whole program.







You can't keep a good bad man down.
READER SERVICE NO. 165

"Attack" begins with a paratroop drop from the Allied helicopter in the background. The Dictator's machine gun at the bottom-center of the display tries to shoot as many soldiers as possible before they overrun the gun emplacement. There are two rows of brick walls which provide cover between death-defying dashes across open terrain, but this is fundamentally a war of attrition. In other words, the Allied commander must expect to lose some men in order to achieve his goal.

The Allies earn one reinforcement for each trooper who reaches either of the two walls closest to the machine gun. The attackers need every man, because the only way to get to the second scenario, "Rescue," is to charge into the teeth of the gun and blow the door behind it. The soldiers can toss grenades at the machine gun nest, but avoiding bullets is the main tactical concern.

After the assault group successfully storms the gun position, the display switches to the "Rescue" phase. The Dictator's arsenal is depleted, so he must prevent the 20 hostages from crossing the courtyard to the waiting helicopters with four oddly assorted weapons: a man on the roof who drops stones on the prisoners, a non-firing tank which runs them down, a truck which shoots at them with a small-caliber machine gun, and a guy who pops out of a trap door and sets landmines. The Allies' machine gun at the bottom of the screen can shoot the various obstacles out of the path of the hostages.

"Escape," the third scenario, is as prosaic as "Rescue" is wacky. It resembles *Choplifter* viewed from an

overhead perspective. The Allies must airlift hostages to freedom in three pairs of choppers despite tenacious defenders on the ground.

"Battle" pits the commander of the Allies against the Dictator in personal combat. The large body of water at the center of the screen separates the rivals, who throw heavy sharpened sticks called poontas at each other.

Each man can move back and forth along his pier, duck or jump to avoid blows, and toss poontas at his foe. A fighter can throw straight by hitting the action button or try a curving toss by moving the joystick at the instant the onscreen figure releases the stick.

This is a fitting climax to this installment of the *Beach-Head* saga, but not half as entertaining as "Attack" or "Rescue." A combatant who scores four hits wins a round and sends the loser into the drink. It's fun once or twice, but players may feel that nine rounds, the length of a complete engagement, is too much.

It's hard to fault either the sound or the graphics. Speech Systems of Berkeley, the folks who did the voice for Activision's *Ghostbusters,* has clearly surpassed itself. The pathetic cries of wounded soldiers in "Attack" are almost too heart-tuggingly realistic. The squeamish can shut off the voice prior to the start of the game.

The visuals are generally outstanding, though "Escape" looks bland next to the other three scenarios. The animation is particularly fine in the opening and closing sequences, and it's hard not to smile at the antics of the little man who pops out of the trap door with the mines in "Rescue."

The four parts of *Beach-Head II* combine into an enjoyable play-experience, even though none is really strong enough to stand alone. It's one of the best head-to-head games for the Commodore, and the computer makes a powerful solitaire opponent.

Who knows? Maybe the Dictator swam away after losing in "Battle" to plague the Allies anew in *Beach-Head III*. Fans of the Carvers' inimitable design style can only hope so.

Access Software, 925 East 900 South, Salt Lake City, UT 84105.

–Arnie Katz

QUESTRON
Strategic Simulations, Inc.
Commodore 64
Disk; $39.95



Warning! Can put bags under your eyes.
READER SERVICE NO. 166

You can't imagine how difficult it was to drag myself away from *Questron* to write this review. It's one of those games that you start playing and suddenly realize it's three hours past bedtime. SSI should label the box "potentially hazardous to your sleep."

If you've ever played *Ultima II,* you have some idea of how *Questron* works. In fact, the game style is licensed under Richard Garriott, *Ultima's* author. You begin the game as a poor peasant kid, fresh off the farm. From there, it's a straight uphill battle to become a knight. Every creature you meet along the way is a potential enemy. Most will attack on sight. But among them there may lurk a few friends. Only an adventurous spirit will ever find out.

You'll have some important considerations to make along the journey, like "Where is your next meal coming from?" It doesn't take long to starve to death out in the wild. Distributed around the countryside are small towns whose businessmen will be happy to take your hard-won gold in exchange for weapons, armor, food, or advice. Of course, this means you've got to get some gold. If you approach the right salesman, you might cut a deal on a horse, whama lama, or raft.

In some of the towns, you can even do a little gambling. I admit to spending a great deal of time in front of the blackjack tables (in real life, one of my few vices). There are also roulette tables and a game called double or nothing. Sorry folks, no ponies or dogs. The closest they get to a greyhound is a Blind Blood Dog that can rip your throat out if you aren't careful.

Hit points is a rather nebulous term for your good health. The more points you have, the more damage you can take in a fight. When the points are gone, so if your life. The people and monsters of *Questron* are not equally dangerous. Some can only inflict minor damage, while others can end your life in a few strokes. The guards that populate the towns and the castle are quite strong. And if you ever make it to the Land of Evil, you'll find out just how tough beasties can be.

What's the point, you ask? The Evil Magician Mantor is threatening to destroy all of Questron and you've got to stop him. With the help of the princess, the king, the doctor and the Wizard Mesron, you might be able to defeat Mantor. But first you have to prove yourself by attaining the rank of knight.

There are a number of puzzles to overcome in order to do well in *Questron*. My only hint would be to talk with as many people (things) as possible. You could pick up something that will be useful down the road.

I can't tell you a couple of the really neat features of the game without giving away some of the puzzles. I will say that *Questron* has incorporated an arcade game and a puzzle that make it more interesting than your average strategy/adventure game. The arcade game tests your dexterity; the puzzle tests your intelligence.

Questron has great graphics and the sound is marvelous. I especially liked the opening theme, despite its lengthening the load time. Be warned, *Questron* takes a long time to load. Also, as you have gathered from my earlier statements, it takes a long time

to play. Even a seasoned adventurer will spend a few days working his way through it. A save feature is included. Four characters can be stored on the disk at one time, so more than one person can be using the same disk.

Whew! Now that this is out of the way, I can get back to that dungeon in the Land of Evil and see what awaits me. The outline effect used to portray the limited visibility in the dungeons is perfect. You really feel like you have to crawl along and watch out for the traps. Ooops! Another trip wire.

Strategic Simulations, Inc., 883 Stierlin Road, Building A-200, Mountain View, CA 94043-1983 (phone: 415-964-1353).

–Cheryl Peterson

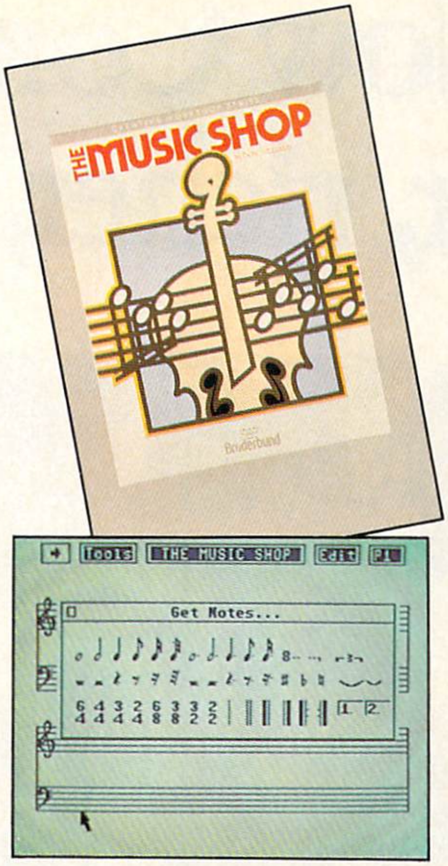
THE MUSIC SHOP
Broderbund Software
Commodore 64
Disk; $44.95

After doing a comparison of a dozen music packages for *Ahoy!* (Nov. '84), I thought I'd seen every configuration imaginable. So when a favorite editor asked me to look at *The Music Shop,* I was prepared to be disappointed. What a pleasant surprise awaited me. Not only does *The Music Shop* have all the great features needed to make it a useful music processor, it emulates the user interface of the Apple Macintosh, almost legendary for its icons, pulldown menus, and dialog boxes.

The Music Shop single-handedly proves my contention that the Commodore's limitations are perpetuated by the programmers who don't make the most of it. In operation, *The Music Shop* functions much as any good Macintosh program. Substitute a joystick for a mouse and off you go.

Any chores that need to be performed are identified by icons at the top of the screen. Pointing to them brings down a menu with highlightable options. The fire button activates the option. What could be simpler?

Even so, I must compliment the documentation. It is attractive, informative, and logically laid out, with sufficient illustrations and tutorial material. In addition, short sections explain a little music theory and how the SID chip works. A handy reference card is included, making it easy to use and learn the keyboard commands available (if you choose to use them instead of the joystick).



Emulates the Macintosh user interface.
READER SERVICE NO. 167

One of last year's better music programs, *Music Construction Set,* used a hand icon–controlled by a joystick –to point at musical symbols that were put onto a staff to create sheet music. Though it functions much the same way, *The Music Shop* uses the pulldown menus to provide a greater variety of options. "Get notes," for instance, brings up a dialog box with notes and rests (whole to 32nd duration), time signatures, bars, repeat symbols, and other musical notations. Another choice in the "Tools" menu allows you to set up the staff groupings, color combinations, and key signatures. One option even allows you to verify the timing of each bar.

Although *The Music Shop* appears to require more steps to operate than *Music Construction Set* or other onscreen music editors, I felt more comfortable with it because the screen didn't look so cluttered. Two double (or four single) staves can be on the screen simultaneously, allowing room for at least four bars of music. Since dialog boxes only appear when they are needed, there is nothing between your and your music.

The editing commands include cut and paste, copy, delete, clear, home, and the most important safety feature, UNDO. While the commands can be entered using the Edit menu, there are two-key combinations that also can be used. When the menu is pulled down, each option is followed by its keyboard command–sort of a subliminal learning experience.

At almost any point in editing, you can play the entire piece to see how you're doing. Though each note is played as it is entered on the staves, it doesn't hurt to occasionally hear the whole thing from the beginning. Pressing the joystick button will stop the playback.

If you aren't happy with the instruments that are playing the music, another menu will allow you to change the three voices. There are eight preprogrammed instruments; or you can design your own by modifying wave forms and envelope data. When you save your song, the instruments are saved as well. Although there are only three voices, you can change each song's instruments up to eight times for each voice.

Though this is enough to make the program functional, *The Music Shop* goes one step further. You can play the music while making the adjustments, so it is easy to judge how effective your modifications are. There are coarse, medium, and fine controls, giving maximum flexibility.

If that's not enough, the program will print out sheet music for you; the whole song in one pass or a page at a time. My only gripe is that it doesn't work with the Cardco printer interface, but if the sheet music meant that much to me, I'd go buy one of the other three interfaces that do work.

Broderbund, 17 Paul Drive, San Rafael, CA 94903-2101 (phone: 415-479-1170).

–Cheryl Peterson

Now Commodore
smarter, richer, more po



64 owners can be pular, and well-traveled.

Here's the on-line service you've been waiting for. You'll love how it saves you time and bucks. How it's a cinch to use. How it runs in color. And how you can try it for $9.95.

Viewtron will make your life much cushier. It'll help you do drudge work like paying bills and shopping. Viewtron also brings information to you on a platter. Things like up-to-the-minute news, financial analysis, weather, sports, software reviews, and an encyclopedia.

Viewtron is backed by the resources of America's leading newspaper publisher. It's the first complete service that runs in color on Commodore 64's. And it's easier to use and easier to read than other services. You don't need to memorize gobbledygook to get at the stuff you want. Just type plain, everyday words like "football scores", "encyclopedia", "stock quotes" or "airfares".

Viewtron costs just 9 cents a minute, weekday nights (after 6 pm) and weekends; 22 cents a minute weekdays.* And unlike other services we don't charge extra if you use a 1200-baud modem.

Smarter.

Viewtron makes you more savvy, and saves you time gathering valuable information. You'll get Commodore software reviews and ratings. Software and hardware advice from experts. Answers to your questions. Current stock prices plus detailed investment analysis. Up-to-the-minute news, and weather forecasts. Continuously updated sports scores. Consumer reports. Movie and book reviews. And use of a very impressive encyclopedia, kept up-to-date.

Richer.

Now you can use your computer to shop for bargains, right from your home. You'll get discounts on top-selling Commodore software. Get big discounts on computer hardware, other electronics, fashion merchandise, gifts, travel, and more. Plus Viewtron has unique electronic auctions where the prices constantly drop instead of rise. The bargains are sensational on products ranging from a Sony Walkman to a sporty red Corvette. In fact, Viewtron subscribers have been saving an average of 44% off retail prices.

More popular.

Viewtron lets you make friends with other Commodore owners in your city, or nationwide. Send messages, join special interest clubs, carry on CB chatter, run classified ads, catch the bulletin boards. How else can you get to know so many nice people who share your interests?

Well-traveled.

Viewtron offers late-breaking vacation bargains for someone ready to travel on short notice. You'll save a bundle. On cruises, air travel, and resort holidays. You'll get car rental discounts. And you can save time before you fly by getting schedule information from our airline bible: the Official Airline Guide. You'll also be able to find the cheapest fares to any particular destination.

For only $9.95, you're off and running.

Send for our software starter kit. Just $9.95. And we'll throw in a FREE hour of Viewtron.** If you don't think Viewtron is the best thing since sliced bread, you can return your starter kit and we'll refund your $9.95. Incidentally, you'll need a major credit card to sign on to Viewtron.

If you need a modem—rejoice! Now you can get Viewtron and a modem for just $39.95. (A $109 value.)

Here's a real honey of a deal. Viewtron, plus a 300-baud Westridge 6420 by Anchor Automation. The modem plugs into your Commodore 64 and has Auto-Dial and Auto-Answer. Or get Viewtron and a high-speed 1200-baud Volksmodem 12, by Anchor, for $189.95. It's a $309 value. Satisfaction is guaranteed. So order now.

Viewtron®

To order call 1 (800) 543-5500. Ask for operator 831. Satisfaction Guaranteed.

Viewtron® is a service of Viewdata Corporation of America, Inc., which is a wholly-owned subsidiary of Knight-Ridder Newspapers, Inc. Commodore 64™ is a trademark of Commodore Electronics, Ltd.

*Viewtron is not available in all cities. Prices may vary in some cities.

**Your free hour is available weekends and weekdays after 6 p.m.

Reader Service No. 212

DISK MAINTENANCE

Reflexive Software
Commodore 64
Disk; $29.95

Disk Maintenance is a group of three utility programs designed to help you get the most out of your Commodore 1541 disk drive. The main program offers the user the ability to change and edit any disk directory at will. The list of commands available with this program is both lengthy and impressive. With a single keystroke users will be able to perform the following functions:

Find the starting and ending address of any file.
View/edit Block Availability Map.
Disassemble any ML file.
Exchange the position of any two files in the directory.
Kill any disk file permanently.
List any file as a BASIC program.
Create a disk menu.
Rename a disk.
Append two BASIC files.
Rename a file.
Scratch a file.
Compare any two BASIC programs.
Validate the disk.
Read a sequential file.
Unscratch a file.
Change a file type.
Protect a file from being scratched.

While I could spend the next 20 pages describing each of these functions, my esteemed editor would, no doubt, frown on my doing so. I will therefore restrain myself and concentrate on but two.

The KILL function listed above is similar to the scratch command that all disk users are familiar with. Unlike a scratched file, however, a killed file can never be recovered. This would be useful if you wanted to give a copy of a terminal disk to a friend, but wanted to be sure that your password file could never be unscratched and used by someone else.

The second function I'll elaborate on is MENU. By using this little gem of a command, it is possible to create a menu of all the programs on your disk that will also load and run each of them. Any program, BASIC or machine language, can be included. With such a menu listed as the first file on each of your disks, loading a program and running it becomes a simple matter of loading the menu with the command LOAD"*",8 and then pressing the key that is indicated next to the program you wish to run. You'll never again have to remember the proper spelling of a filename or the correct SYS to use with a machine language program!

ABLE, the second program in this package, is an Advanced Block Editor that will allow the user to view and/or alter the contents of any block on the disk. Included in *ABLE* are functions for working with sprites directly on the disk, numerous help screens, and the ability to enter information as decimal numbers, hex numbers, ML instructions, and BASIC tokens. *ABLE* also supports the use of undocumented opcodes which makes machine code editing a breeze.

Compactor/Uncompactor, the final program, performs two functions. The first is compacting a BASIC program. This is equivalent to shortening the program because each line will be filled to its maximum of 250 bytes. The compactor will also remove all REMs and any leading colons it encounters. The second function will take any BASIC program and expand it by putting each individual command on its own line. This makes debugging much easier since a line can be changed without worrying about affecting other commands on the same line.

Disk Maintenance is, quite simply, one of those programs that no serious Commodore user should be without. Its speed, versatility, and friendliness have earned this program a permanent spot, not on my shelf, but alongside my computer where it will always be within easy reach.

Reflexive Software, 6507 39th Ave N.E., Seattle, WA 98155 (phone: 206-525-6873). *—B.W. Behling*

CARTRIDGE EXPANDER

Navarone Industries, Inc.
C-64, C-128
$24.95

The Cartridge Expander from Navarone Industries is a handy gadget that allows you to have up to three ROM cartridges plugged into your C-64 or C-128 at once. Through the use of a three-position slide switch, you can change from one cartridge to another without the hassle of turning off the computer and removing the present cartridge and replacing it with another. This not only saves you time by keeping your most used cartridge in one place, but also reduces the amount of wear and tear your cartridge port would normally receive. Being able to rapidly switch from one ROM cartridge to another might not seem like a big deal to many people but those of us who have paid $60.00 to $70.00 repair charge for a new cartridge port connector can readily appreciate the advantage of owning this particular piece of equipment!

Another nice feature of this device is its push-button reset switch that allows you to return your computer to its power-up state without any loss of memory. By combining the use of this switch with a program to recover accidentally newed programs, you need never fear losing a program by the inadvertent use of the "NEW" command. This tiny switch is also great for breaking out of an ML program that has "crashed" or otherwise "locked up" your computer.

Although there are similar devices on the market for Commodore computers, the $24.95 price tag of this model places it easily within the reach of most users.

Navarone Industries, Inc., 19968 El Ray Lane, Sonora, CA 95370 (phone: 209-533-8349).

—B. W. Behling

FOREWORD

Experienced users are well aware of the need to maintain backup copies of important disk files. It only takes a single incident of "diskus destructus" to turn the most complacent operator into a dedicated archivist. Even software purveyors with the most complex of copy protection schemes stress the importance of maintaining backup copies of all disk files. The reasons are simple. It takes very little effort to permanently damage the mylar film (only 3/1000 of an inch thick) and its microscopic magnetic coating on which digital data is stored. The typical computer's environment is teeming with forces hostile

to the well-being of the floppy disk, the most common of which are dust, smoke, magnetic fields, and temperature extremes; not to mention the ever-present possibility of physical damage due to improper handling.

In spite of this universal need for backup copies, early 1540/1541 disk drive users were severely handicapped when it came to backing up disk files. Although the Commodore DOS has a built in COPY command, it is of very little value for a single drive system. Having multiple copies of a file on single disk does very little to enhance data security. The only backup utility on the Test/Demo disk supplied with the disk drive (Jim Butterfield's *Copy/All*) required two disk drives for its use. Even then it took half an hour to back up a single disk.

It wasn't until the coming of the *Commodore 64 Disk Bonus Pack* that single drive users obtained some relief. The *1541 Backup* program on that disk allowed for a full disk backup with a single drive. Of course it did require six pairs of disk swaps and about 40 minutes to copy a full disk.

Nevertheless, we old-timers persevered. We gritted our teeth and hung in there, meanwhile inventing ingenious ways to keep ourselves amused while waiting for a disk to be copied. Of course, the 1541 wasn't too happy with this state of affairs. It frequently made its displeasure well known by demanding periodic alignment of its read/write head, as well as other chronic complaints. Ultimately our patience was rewarded. Third party software vendors strived to fill the gap which Commodore has left open to this very day. Both commercial and public domain copy programs have been developed which greatly reduced the time required for the copy process. The very latest of these offerings, which we will look at, gives very impressive performance.

An inevitable side effect of creating a high speed copy utility for the 1541 is the ability to reproduce many of the existing copy protection schemes. Although it would be feasible to omit these features from a copy utility, the publishers rarely do so. It has been found to make good economic sense to promote a copy program with these capabilities included.

Copy protection, piracy, and the software buyer's rights are regular topics in the editorial pages of every computer publication. The discussions appearing in both national publications and local newsletters have ranged hot and heavy under, over, and all around the fence. In the midst of all the controversy one thing remains abundantly clear. The advertising for products designed to assist the user in backing up his software continues unabated. Furthermore, in what seems to be an otherwise depressed market, these products continue to command premium prices. This is in spite of the fact that many of these products are rapidly obsoleted as software producers improve their copy protection schemes.

At the risk of being outdated before these reviews are published, we present two products which as of this writing represent the state of the art in their respective fields for the C-64. We feel that both of these utilities have sufficient legitimate purpose to remain of value for some time to come. This is in spite of the rather colorful names under which they are being marketed.

FASTHACK'EM
Basement Boys Software
Commodore 64
Disk; $29.95

Based on physical appearances, *FASTHACK'EM* certainly lives up to its company's name. Packaging and documentation is minimal, consisting of a custom printed cardboard disk mailer and a single sheet of instructions. The brief documentation is further emphasized by the warranty and copyright notices which make up more than 60% of it. Fortunately, the program is menu driven and easy to use. Most users will require a minimum of experimentation to discover the program's capabilities. In addition, better documentation has been promised for the final release.

FASTHACK'EM is the product of Mike J. Henry. For newcomers in the audience, Mr. Henry was one of the people behind *Disector,* one of last year's most successful copy programs, as well as a powerful and still pertinent disk utility. Apparently Mr. Henry has decided to take his disk drive expertise and branch out on his own. The results are quite impressive. *FASTHACK'EM* is probably the fastest way to copy an entire 1541 formatted disk at the present time.

Actual performance depends on your system configuration. With a single 1541 disk drive, copying time for an unprotected disk is on the order of two minutes. Actual results will depend on the manual dexterity of the user, as three pairs of disk swaps are required. With two disk drives, a 35-second unverified and 56-second verified copy is possible: a truly impressive performance. Users of a single 1541 also have the option of copying many forms of protected software with a nibble copy utility.

If this is not enough, *FASTHACK'EM* provides an auto backup feature. In this mode, two 1541 disk

drives are set up to talk to each other all on their own. You can even pull the disk drive serial port cable and turn off the computer once the process has begun. What's more, multiple copies are possible. The drives will sense the removal and insertion of the source and destination diskettes. Some manual dexterity is required, as copying automatically starts three seconds after the destination diskette is inserted into the drive. It is of course up to the user to maintain the proper sequence of disk removals and insertions. To further speed things up, the auto backup utility also lets you select a range of tracks (from 1 to 40) for copying.

Owners of the MSD SD-2 will be immensely pleased with *FASTHACK'EM*. Two full disk backup versions are provided for this drive. These result in a 60-second and 68-second backup of an entire disk, nearly twice as fast as the MSD SD-2 can do on its own. In addition, these utilities allow the SD-2 to copy many types of protected software. The SD-2 copy routines also allow the setting of a range of tracks from a minimum of track 1 to a maximum of track 40.

Copy protection schemes have matured considerably since the introduction of the 1541. Current methods no longer bump the disk head mechanism against the track one stop, a contributing cause to 1541 alignment problems. Many of these schemes are rather sophisticated, employing such esoteric procedures as variable density recording on a single track, half tracking, and recording on normally unused tracks beyond track 35. As a result, it is very difficult if not impossible to design a general purpose copy utility for these schemes. A number of programs have attempted to handle this problem with limited success. These generally have required the user to provide several disk parameters which describe the protection to be duplicated. The problem with this approach is the difficulty involved in discovering the proper parameters.

The *FASTHACK'EM* nibble utility has some limited parameter input capabilities. However, the user is left totally on his own as to their proper application. Instead, taking the bull by the horns, a parameter copier is included which lists 17 specific programs and a single major publisher by name. Actually the copy utility sometimes cheats. No attempt is made to duplicate the original protection scheme of all of these programs. Instead an unprotected copy is created by modifying the original program code right on the disk. The programs represented are all currently popular titles. We expect the exact roster to vary as time goes on. We leave you to your own means in finding out if your favorite program is represented.

A high speed utility loader is part of the *FASTHACK'EM* program. This seven-block routine may be easily copied for your own use. Installation in the C-64's $C000 block is simple. The result is a better than six to one improvement in the LOADing times of program files. Other disk operating times are not affected.

FASTHACK'EM is a much-needed utility for Commodore disk users. As of this writing it is the fastest whole disk backup utility we know of. Two single 1541 disk drives can now back up a disk in less time than was formerly possible with a dual disk drive. Even the performance of the MSD SD-2, the mainstay of the cottage industry Commodore publisher, has been improved by nearly 100%. User groups across the country have already recognized *FASTHACK'EM*'s capabilities in making real-time copies of public domain libraries during club meetings.

Of course there are some disadvantages. The demands on the user are now greater than ever before. Very little time remains between disk swaps for writing disk labels or other mundane activities. Gone are the leisurely coffee breaks or refrigerator raids while waiting for a 1541 to do its stuff. Insted the user is tied down to the machine, feeding it disks at an ever increasing rate. Most users will gladly make the sacrifice.

Basement Boys Software, P.O. Box 30901, Portland, OR 97230-0901 (phone: 503-256-5506).

—Morton Kevelson

ISEPIC
Starpoint Software
Commodore 64
$64.95

In a related but independent event, Chip Gracey in conjunction with *Starpoint Software* (publishers of *Disector,* mentioned above) have released *Isepic* (pronounced ice pick) for the Commodore 64. *Isepic* is a plug-in cartridge which at the flick of a switch replaces the Kernal ROM with its own operating system. This has the effect of literally stopping the microprocessor in its tracks and removing control from whatever program is running at the time. *Isepic* then lets you save the entire contents of memory to disk. This includes the full contents of the 64 kilobytes of RAM as well as the status of the microprocessor registers and the computer's I/O registers. The resultant "snapshot" is in a form suitable for reLOADing by *Isepic*, at which time the originally executing program picks up at the precise moment at which it left off.

The *Isepic* cartridge contains two kilobytes of static RAM, several digital logic chips, and a single switch. Before booting the source program, the cartridge RAM is LOADed with the *Isepic* operating system. A flick of the cartridge switch then hides the *Isepic* cartridge by electronically disconnecting it from the expansion port. The computer is then reset to the power up state, at which time the subject software is booted up in the appropriate fashion.

The *Isepic* operating system lets you do several things before you relinquish control back to the C-64. You can view the disk directory, format a disk, and create a special *Isepic* disk for use during the snapshot process. For advanced users a memory editor and mini disassembler are included. At the time of flicking the switch, *Isepic* also fills all of memory with a predetermined pattern of its own choosing. The purpose of this will be seen shortly.

The ramifications of all this activity are startling, to say the least. Most significant is the fact that this treatment will bypass any disk copy protection schemes which may have been used by the original software boot routines. Backup copies can thus be made of the resulting snapshot files. However, in the original snapshot form, the *Isepic* cartridge and software is required to reboot the original program. To get around this, *Isepic* lets you "break" the program.

For those unfamiliar with this term, breaking a program refers to the process whereby the true hacker disassembles a copy protected program and removes the original protective code. The origin of the term probably lies in the activity burglars perform in the still of the night. A properly broken software package can be easily archived. Very often, operating convenience in the form of shorter loads is enhanced as well.

Although we have no definitive proof, it is generally believed that true hackers have no real interest in the programs they are breaking. Their primary concern lies in what is perceived as the challenge of discovering the secrets of the code placed into the program by the original authors. Some original software authors even encourage this behavior by placing hidden messages on the disk or in the code. Very often these are phrased as a dare to the hacker, in effect challenging him to do his worst.

Of course, most major software publishers take a very dim view of these activities and generally wish it would all just go away. As we mentioned above, there are many aspects to this situation. We can only encourage our readers to let their consciences guide them along the proper path.

As applied to *Isepic*, breaking the program is a compacting process which results in a stand alone LOADable file which can RUN without the additional benefit of the *Isepic* package. To accomplish this, *Isepic* makes use of the pattern previously placed in RAM as mentioned above. This allows unused parts of the snapshot to be eliminated from the final disk files. Also, *Isepic* looks for any repetitive patterns in the program code for possible additional compaction.

Not all snapshots can be broken by *Isepic*. A small amount of memory has to be made available for the stand alone boot routines which are included with the broken program. These routines reconstruct a RUNable image after the program is LOADed. In the tradition of the true hacker, these routines also display the *Isepic* logo while booting the program.

Isepic will convert many copy protected programs to easily archivable form. Many broken programs will LOAD faster due to the elimination of the initial opening screens and menus associated with most commercial software. Of course, some protected programs have their own built-in speedup routines which will be negated by the *Isepic* treatment. *Isepic* supplies its own boot utility which is incorporated in every broken program. A typical *Isepic'd* program is on the order of 35 kilobytes. These LOAD in less than 30 seconds with the *Isepic* speedup routines.

Isepic will not remove the protection from software which relies on security keys or dongles for copy protection. Of course, such programs do not require the attention of *Isepic* in the first place. *Isepic* will not successfully remove the protection from programs which check the disk at regular intervals during normal operation. These programs will still require the original protected disk for proper operation.

The *Isepic* process is fairly lengthy. It takes about five minutes to create the initial snapshot. A similar interval is needed to perform the break operation. Additional overhead is consumed by disk formatting and LOADing the original program. It should take about 30 minutes for a complete *Isepic* procedure.

Once again we caution our readers that *Isepic'd* programs are still copyrighted by the original authors and publishers. In addition, there is the matter of the copyright on the *Isepic* code itself. Illicit distribution of copies of *Isepic'd* programs may bring down the wrath of the original publishers, as well as Starpoint Software

and Scott Blum, the official Starpoint Cynic.

Starpoint Software, Star Route 10, Gazelle, CA 96034 (phone: 916-435-2371). *—Morton Kevelson*

EASY GUITAR

DJ Software
Commodore 64/128
Disk; $39.95

In 1955, the image of Elvis rocking and reeling on TV provoked thousands of teenage boys to learn to play guitar. You probably saw more than a few of those boys—grownup now—at the recent Live Aid extravaganza. This "Electronic Woodstock" may inspire the *next* generation of guitar players the same way Elvis did the last. In 1985, however, we've got something Townsend, Clapton, McCartney and the rest never had—a Commodore 64/128 and a program called *Easy Guitar.*

There are dozens of software packages that teach us how to read music, write music, and create music. Oddly, there are only a few that actually teach us how to *play* a musical instrument. And *Easy Guitar* may be the only one that teaches the instrument most people want to learn to play. It was conceived and programmed by David Perry, a Seattle-based musician and record producer.

The program follows a very simple step-by-step method. Beginning players are shown a graphic labeling the parts of a guitar, and the computer generates the six tones needed to tune the strings. (Unlike most of us, the C-64/128 has perfect pitch.) A short lesson in music theory follows, but you don't need it to learn how to play beautiful music. The Beatles, among others, proved that.

The computer is a good guitar teacher because it can do more than books or audio tapes. A book can show you where to place your fingers on the frets. A tape can let you hear what it should sound like. But the computer can do both—and at the same time. In demonstrating how to play scales, for instance, you see the correct fingering and hear the scale being correctly played. A guitar teacher can do this too, but try to find a *human* who will teach you how to play guitar for a flat fee of $39.95.

The program contains diagrams of the 50 most common chords you'll run into, which can be practiced in over a dozen different progressions. During practice sessions the computer plays the chord changes with you, backed by a bass and drum simulation called "The Commodore 64 Back-Up Band." You can set the band to play as fast or slow as you'd like with a built-in metronome that clicks on each beat.

Easy Guitar claims to be for both beginners and advanced players, but it's really only a tool for the complete novice guitar player. There's no mention of fingerpicking at all, and hardly any guidance on strumming or what to do with the hand that strikes the strings. This doesn't mean it's bad, just that people who already play well won't get much out of it.

Only one chord diagram fits on the screen at once, and the program suggests buying a regular chart illustrating the common chords. I thought the designers could have easily included one in the package. The program could also have been improved by including some synthesized pop tunes for users to play along with the computer. The designers chose to include a multiple-choice test and a "Catch a Chord" video game instead, both of which seem like filler.

Still, as a guitar player myself, I found *Easy Guitar* to be the perfect way for people who have never played a note to get their feet wet. The lessons are clear, simple, and take advantage of the computer's interactive capabilities. You can learn at your own pace. You don't have to plow through a thick instruction manual—in fact, instructions aren't even included in the program. Everything you need to know is right up there on the screen.

For the kids who caught the guitar bug watching Live Aid, enjoy your guitar. It's an easy instrument to learn, but a hard one to master. You probably won't see this program in your local store, but you can make an order or get more info by writing DJ Software, 10636 Main St., Suite 414, Bellevue, WA 98004.

—Dan Gutman

GIANT PRINTER SALE

10"/15½" COMSTAR 10X/15½X 120-140 CPS

COM-STAR PLUS + (Tractor-Friction Printer)

"Premium Quality" **$179.00**

- Tractor Friction Printer
- Dot Matrix, Impact, Prints Single Sheets or Continuous Feed Paper, 10" or 15½" Carriage
- Print buffer
- 9 x 9 Dot Matrix, Double Strike
- Near Letter Quality, High Resolution, Dot Bit Image
- Underlining, Left-Right Margin
- True Low Descenders, Super and Subscript
- Prints Standard, Block Graphics, Italics
- Centronics Parallel Interface



COM-STAR 10X List $399 **Sale $179**
COM-STAR 15½X List $599 **Sale $249**

One Year Free Replacement Warranty

PRINTER
Impact Dot Matrix

PRINT HEAD
9 Wire (User Replaceable)

PRINT SPEED — Bi-directional Logic Seeking
Super 10X 120-140; Super 15½X 120-140

PAPER SLEW SPEED
10 LPS at 1/6" Spacing

PRINT BUFFER
One Line Standard

PRINT SIZE
10, 12, 17, 5, 6, 8.5 CPI

NUMBER OF COLUMNS
Super-10X: 80, 96, 136 (40, 48, 68 Double Width)
Super-15½X: 136, 164, 232 (68, 82, 116 Double Width)

CHARACTER MATRIX
9 x 9 Standard, with True Descenders
18 x 9 Emphasized
18 x 18 Double Strike
6 x 6 Block Graphics
60 x 72 Low Resolution, Bit Image Graphics
120 x 144 Hi Resolution, Bit Image Graphics
240 x 144 Ultra Hi Resolution, Bit Image Graphics

CHARACTER SETS
96 Standard ASCII, 32 Block Graphic Characters, 96 Italics Characters

CHARACTER FONTS
Normal (10 CPI)
Elite (12 CPI)
Condensed (17 CPI)
Enlarged (5, 6, 8.5 CPI)
Emphasized
Double Strike
Super and Sub Script

SPECIAL FEATURES
Self-Test
Macro Instruction
Continuous Underline
8 Bit Selectable Interface
Column Scan Bit Image Graphics
Vertical and Horizontal Tabs
Skip Over Perforation
15.5" Carriage (Super-15½X Only)

LINE SPACING
1/6", 1/8" Plus Programmable by n/144" or n/72"

PAPER HANDLING

	SUPER-10X	SUPER-15½X
Roll Paper:	8.5" to 10" Wide	8.5" to 15" Wide
Cut Sheet:	8" to 10" Wide	8" to 15" Wide
Copies:	3 Carbonless Sheets	3 Carbonless Sheets

RIBBON
Super 10X Cartridge Ribbons $9.95, Super 15½X Cartridge $14.95

INTERFACE
Standard Parallel (Centronics Compatible)

DIMENSIONS — in inches & WEIGHT — in lbs.
SUPER-10X
14W X 10-9/16D x 2 7/8H, 11 lbs.

All product data subject to change without notice.

Parallel Interfaces: Commodore 64 — $39.00 Apple II, IIe, II+ — $59.00 Atari — $59.00

Add $10.00 ($14.50 for 15½" Printers) for shipping, handling and insurance. Illinois residents please add 6% tax. Add $20.00 for CANADA, PUERTO RICO, HAWAII, ALASKA, APO-FPO orders. Canadian orders must be in U.S. dollars. WE DO NOT EXPORT TO OTHER COUNTRIES, EXCEPT CANADA.

Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail!

VISA — MASTER CARD — C.O.D. No C.O.D. to Canada, APO-FPO.

PROTECTO

We Love Our Customers

22292 N. Pepper Rd., Barrington, Illinois 60010

312/382-5244 to order

Reader Service No. 208

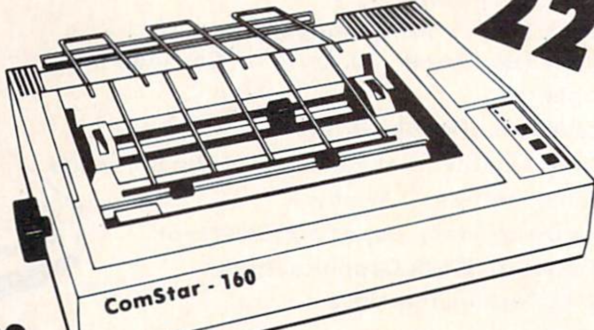
GIANT PRINTER SALE

10"/15½" COMSTAR 160 — 150-170 CPS

COMSTAR 160 + High Speed (Tractor-Friction Printer)

$229.00

- Tractor Friction Printer
- Dot Matrix, Impact, Prints Single Sheets or Continuous Feed Paper, 10" or 15½" Carriage
- Print buffer
- 9 x 9 Dot Matrix, Double Strike
- Near Letter Quality, High Resolution, Dot Bit Image
- Underlining, Backspacing, Left-Right Margin
- True Low Descenders, Super and Subscript
- Prints Standard, Block Graphics
- Better than Gemini 10X
- Centronics Parallel Interface



10" COMSTAR 160 List $499 **Sale $229**

15½" COMSTAR 160 List $699 **Sale $299**

One Year Free Replacement Warranty

PRINTER

Impact Dot Matrix

PRINT HEAD

9 Wire (User Replaceable)

PRINT SPEED — Bi-directional Logic Seeking

10" ComStar 160: 150-170 CPS; 15½" ComStar 160: 150-170 CPS

PAPER SLEW SPEED

10 LPS at 1/6" Spacing

PRINT BUFFER

2K

PRINT SIZE

10, 12, 17, 5, 6, 8.5 CPI

NUMBER OF COLUMNS

10" ComStar 160: 80, 96, 136 (40, 48, 68 Double Width)
15½" ComStar 160: 136, 164, 232 (68, 82, 116 Double Width)

CHARACTER MATRIX

9 x 9 Standard, with True Descenders
18 x 9 Emphasized
18 x 18 Double Strike
6 x 6 Block Graphics
60 x 72 Low Resolution, Bit Image Graphics
120 x 144 Hi Resolution, Bit Image Graphics
240 x 144 Ultra Hi Resolution, Bit Image Graphics

CHARACTER SETS

96 Standard ASCII, 64 Special, 32 Block Graphic, 96 Downloadable, 88 International Characters

CHARACTER FONTS

Normal (10 CPI)	Emphasized
Elite (12 CPI)	Double Strike
Condensed (17 CPI)	Super and Sub Script
Enlarged (5, 6, 8.5 CPI)	

SPECIAL FEATURES

Self-Test	15.5" Carriage (15½" ComStar 160 Only)
Macro Instruction	
Continuous Underline	
8 Bit Selectable Interface	
Column Scan Bit Image Graphics	
Vertical and Horizontal Tabs	
Skip Over Perforation	

LINE SPACING

1/6", 1/8" Plus Programmable by n/144" or n/72"

PAPER HANDLING

	10" COMSTAR 160	15½" COMSTAR 160
Roll Paper:	8.5" to 10" Wide	8.5" to 15" Wide
Cut Sheet:	8" to 10" Wide	8" to 15" Wide
Sprocket:	3" to 10" Wide	5" to 15.5" Wide
Copies:	3 Carbonless Sheets	3 Carbonless Sheets

RIBBON

10" ComStar 160 Cartridge $9.95, 15½" Comstar 160 Ribbon $14.95

INTERFACE

Standard Parallel (Centronics Compatible)

DIMENSIONS — In inches

10" COMSTAR 160	15½" COMSTAR 160
15.36W X 12.6D x 5H	23.4W x 14.3D x 5.2H

WEIGHT — In pounds

10" COMSTAR 160: 11 lbs. 15½" COMSTAR 160: 18.7 lbs.

All product data subject to change without notice.

Parallel Interfaces: Commodore 64 — $39.00 Apple II, IIe, II+ — $59.00 Atari — $59.00

Add $10.00 ($14.50 for 15½" Printers) for shipping, handling and insurance. Illinois residents please add 6% tax. Add $20.00 for CANADA, PUERTO RICO, HAWAII, ALASKA, APO-FPO orders. Canadian orders must be in U.S. dollars. WE DO NOT EXPORT TO OTHER COUNTRIES, EXCEPT CANADA.

Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail!

VISA — MASTER CARD — C.O.D. No C.O.D. to Canada, APO-FPO.

PROTECTO

We Love Our Customers

22292 N. Pepper Rd., Barrington, Illinois 60010

312/382-5244 to order

Reader Service No. 208

COLOR MONITOR

SALE!!!

Sale

16 COLORS

(Premium Quality)

- Built in Speaker and Audio
- Front Panel Controls
- For Video Recorders
- For Small Business Computers
- Apple - Commodore - Atari - Franklin - etc.

Add $14.50 shipping, handling and insurance. Illinois residents please add 6% tax. We do not ship to foreign countries (including Canada), Alaska, Puerto Rico, APO-FPO.
Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail!



13" Color Computer Monitor

*Connecting cable $9.95

(Premium Quality)

- Beautiful Color Contrast
- High Resolution
- Separated Video
- Sharp Clear Text
- Anti-Glare Screen
- 40 Columns x 24 Lines
- Supports 80 Columns

List $329.00

Sale $169.95*

13" RGB & COMPOSITE MONITOR
List $399.00

Must be used to get 80 columns in color with 80 column computers (C128 & Atari ST). (add $14.50 shipping)

Sale $279.00*

14" COMMODORE 1702 COLOR MONITOR
(add $14.50 shipping) List $299.00

Sale $179.00*

15 Day Free Trial - 90 Day Immediate Replacement Warranty

12" XTRON SUPER HI-RESOLUTION MONITOR

80 Columns x 24 Lines, Super Hi-Resolution 1000 lines Green or Amber Super-Clear "Easy to Read" text with special anti-glare screen! *(Ltd. Qty.)* List $249.00

Sale $109.95*

12" ZENITH HI-RESOLUTION GREEN OR AMBER TEXT DISPLAY MONITOR

80 Columns x 24 Lines, Hi-Resolution, crisp clear easy to read text with anti-glare screen! A MUST for word processing. *(Ltd. Qty.)* List $199.00

Sale $89.95*

12" MONITOR GREEN OR AMBER TEXT
List $159.00

80 Columns x 24 Lines, easy to read up front controls *(Ltd. Qty.)*

Sale $69.95*

* Connecting cable $9.95

• LOWEST PRICES • BEST SERVICE IN U.S.A. • ONE DAY EXPRESS MAIL • OVER 500 PROGRAMS • FREE CATALOGS

Add $10.00 for shipping, handling and insurance. Illinois residents please add 6% tax. Add $20.00 for CANADA, PUERTO RICO, HAWAII, ALASKA, APO-FPO orders. Canadian orders must be in U.S. dollars. WE DO NOT EXPORT TO OTHER COUNTRIES, EXCEPT CANADA.
Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail!
VISA — MASTER CARD — C.O.D.
No C.O.D. to Canada, APO-FPO.

PROTECTO

We Love Our Customers

22292 N. Pepper Rd., Barrington, Illinois 60010

312/382-5244 to order

Reader Service No. 208

SAVE

Economy* to Arcade Quality

JOYSTICK

Sale

Low As

*$8.95

SALE

Low As

*$8.95

Ultimate Arcade Quality Model

Commodore 64
VIC 20
PLUS All Atari Machines

Coin Controls 5000



Unconditional 2 Year Warranty

List $24.95

Sale $16.95

Rated No. 1

5200 Joystick
Allows keypad hook-up
List $29.95 **Sale $24.95**

Professional Cadillac Model

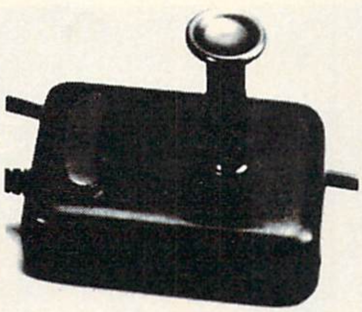


List $19.95

Sale $12.95

"Three Way Firing" Options

*Single Button Economy Model



List $12.95

Sale $8.95

Add $3.00 for shipping, handling and insurance. Illinois residents please add 6% tax. Add $6.00 for CANADA, PUERTO RICO, HAWAII, ALASKA, APO-FPO orders. Canadian orders must be in U.S. dollars. WE DO NOT EXPORT TO OTHER COUNTRIES, EXCEPT CANADA.
Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail!
VISA — MASTER CARD — C.O.D.
No C.O.D. to Canada, APO-FPO.

PROTECTO
We Love Our Customers
22292 N. Pepper Rd., Barrington, Illinois 60010
312/382-5244 to order

Reader Service No. 208

Commodore 64

Computer Learning Pad

- **Makes Graphic Tablets Obsolete**
- **Includes Special Printer Feature**

Sale! $37^95

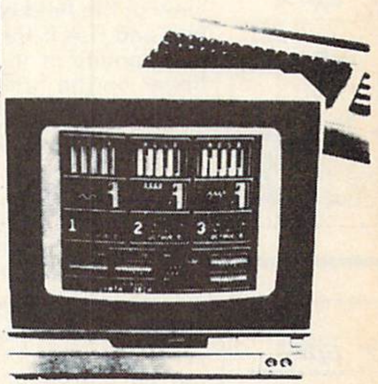
Hi-Resolution!



Now you get this **Fantastic Tech Sketch Computer Light Pen Program** with a **Professional Light Pen Free!** ($39.95 value) plus the **Micro Illustrated Graphics Pen Program** that allows you to draw on your T.V. or monitor screen (better than Gibsons $99 light pen). Whatever you can draw on the screen you can print on your printer (A Tech Sketch Exclusive.) (Disk) List $59.95 **Sale $37.95.**

Also available is **Lite Sprite**, a light pen driven sprite builder (List $39.95 **Sale $29.95**).

MUSIC PORT

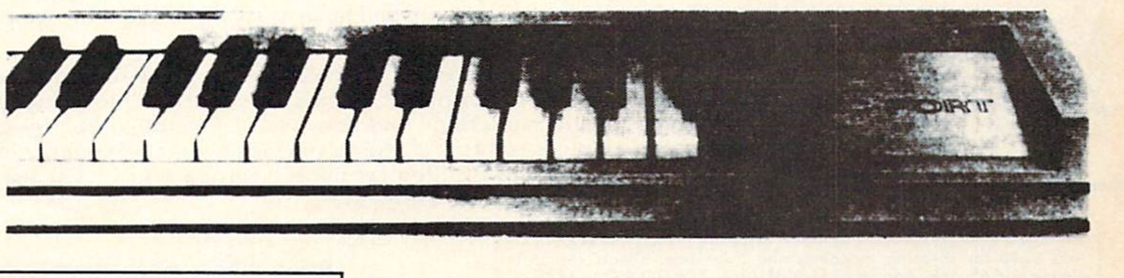


The ultimate music synthesizer and multi-track recording system for the Commodore 64. A **Full-sized REAL Keyboard** and all software including preprogrammed songs are included. (Disk) List $149.95 **Sale $99.00.**

Fully Responsive "MUSICIANS" Keyboard

Sale! $99^00

No Extra Software Required



Add $3.00 for shipping, handling and insurance. Illinois residents please add 6% tax. Add $6.00 for CANADA, PUERTO RICO, HAWAII, ALASKA, APO-FPO orders. Canadian orders must be in U.S. dollars. WE DO NOT EXPORT TO OTHER COUNTRIES, EXCEPT CANADA.
Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail!
VISA — MASTER CARD — C.O.D.
No C.O.D. to Canada, APO-FPO.

PROTECTO
We Love Our Customers
22292 N. Pepper Rd., Barrington, Illinois 60010
312/382-5244 to order

Reader Service No. 208

Commodore 64 **Commodore 64**

BEST VALUE ***BEST VALUE***

EPYX™

The Best Arcade Games Around



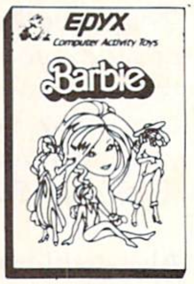
Winter Games — Experience the challenge of six winter sporting events. In the Bobsled, you're right in the tube careening along the walls. At the SkiJump, you control your form in take-off, flight and landing. In Figure Skating, timing counts for the transitions, the jumps and landings. Choreograph your own routine in Free Style Skating. Hot Dog Aerials push your agility to new heights. The Biathlon challenges your endurance in cross-country skiing. There's even an opening ceremony, complete with national anthems. One to Eight Players. (Disk) List $39.95. **Sale $22.95.**



The World's Greatest Football Game — This one has it all — strategy, action, three views of the field. It's in a class all its own! Finally, a football game that not only puts you on the field, but also on the sidelines in the coach's shoes. Use the "Playbook" or design your own offensive and defensive plays. Then, grab the joystick and put your strategy to the test. You control key players to run a sweep, make a tackle, throw a pass and even kick a fieldgoal. All the action and all the strategy make this your favorite football game. (Disk) List $39.95. **Sale $22.95.**



Jet Combat Simulation. Flight simulation programs are the most requested in the country. The key to a good one is realism, the sensation of being in the cockpit — guiding the plane through take-offs, landings and air to air — air to ground combat. You are an Air Force pilot and your mission is critical. Your success in completing your orders depends on how quickly and accurately you react. Very Intense — Fantastic Graphics, animation and control elements. (Disk) List $39.95. **Sale $22.95.**



Barbie — The biggest name in dolls. Browse through Barbie's closet full of beautiful clothing and dress her for the party. You can cut her hair or make it longer and color it or change the style. Buy new clothing at the boutique or any of the six other specialty shops, or even go to the dress shop and create Barbie's designer clothes with the help of the computer. The combinations are endless and so is the fun. One or two players. (Disk) List $39.95. **Sale $24.95.**



Ballblazer — Unique split-screen, 3-D graphics give you and your opponent a first person view of the field of play. You race across the playfield in your Rotofoil trying to capture the ball and fire it through the goal before your opponent. The winner is the player with the most points at the end of the timed competition. Hold onto your joystick and keep that finger on the fire button, this is the type of two player head-to-head action you've been waiting for. Two Players. (Disk) List $29.95. **Sale $24.95.**



Summer Games II — The original Summer Games was last Summer's No. 1 seller. Carry on the tradition with another chance to "Go for the Gold!" Introducing Summer Games II with 8 new Olympic events including Kayaking, cycling, fencing, diving, track & field, gymnastics and equastrian. The excitement of Olympic competion is present in this new version as it was in the original. Great graphics and sound effects. This one's a winner! (1 to 8 players). (Disk) List $39.95. **Sale $24.95.**



G.I. Joe — The best selling toy soldier is now available as an activity toy on the home computer. Select the battle situation then choose the equipment you think you will need to get the job done. Choose from a number of uniforms and weapons in your well stock arsenal and get ready for the action. Play alone or with a friend, if you plan the right strategy you will complete the mission if not you will have to try again. One or two players. (Disk) List $39.95. **Sale $24.95.**



Rescue On Fractalus! — Your mission is to fly your Valkyrie Fighter through the Jaggi defenses and rescue the downed Ethercorps pilots. Sounds easy, but don't let it fool you. It's tough enough just to navigate the mountains and canyons of Fractalus, but try doing it while destroying enemy gun emplacements or dodging suicide saucers. We supply the Long Range Scanner, Dirac Mirror Shield and Anti-Matter Bubble Torpedoes ... YOU supply the skill and guts! One Player. (Disk) List $29.95. **Sale $24.95.**

Add $3.00 for shipping, handling and insurance. Illinois residents please add 6% tax. Add $6.00 for CANADA, PUERTO RICO, HAWAII, ALASKA, APO-FPO orders. Canadian orders must be in U.S. dollars. WE DO NOT EXPORT TO OTHER COUNTRIES, EXCEPT CANADA.

Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail!

VISA — MASTER CARD — C.O.D.

No C.O.D. to Canada, APO-FPO.

PROTECTO

We Love Our Customers

22292 N. Pepper Rd., Barrington, Illinois 60010

312/382-5244 to order

Reader Service No. 208

Sale
LOW AS
$29.95
List $39.95

Program Your Commodore 64 In Plain English
(No need to know Basic)

CodeWriter

New Program Generators!!

FileWriter — lets you control information so you can enter it, find it and change it, just the way you'd like. Simply draw any kind of screen you want to handle your data whether its information about bills, taxes, recipes, club memberships, stamp collections or your new home business. It will calculate automatically, hold grand totals, test for bad information for one form or hundreds. There is no need to "program" or know anything about computers. Most of the instructions you'll need are right on the screen for you. (Disk) List $39.95. **Sale $29.95.** *Coupon $24.95.*





ReportWriter — Lets you make 'data' into information that can be reorganized and presented to help you make decisions. You need invoices turned into commission reports, customer leads tuned into mailing labels, accounts receivable turned into letters to your customers. ReportWriter can handle all of these requests. You create your own report ideas with ReportWriter. A report can be simple columns of figures, a memo, a check, a label, whatever you need. You may create as many reports as you like from the same files; it will report to the computer screen or to your printer. (Disk) List $39.95. **Sale $29.95.** *Coupon $24.95.*

MenuWriter — MenuWriter links programs together so the user can select the program he needs by pressing just one key. Where good program design calls for several different files or reports to load smoothly and quickly, MenuWriter does the job. You set up screen after screen (up to 21) of simple clear menu formats to link your programs together. Menus can call programs to auto-load or simply call other menus. You control the whole process -from color selection to screen layout, to key choice. Whether you're working with CodeWriter programs or your own, MenuWriter will give you professional menu results the first time. (Disk) List $39.95. **Sale $29.95.** *Coupon $24.95.*





SpeedWriter — What do you do with a BASIC program that you wrote (or bought, or typed) that is nice, does what you wanted it to do, but is just too slow to suit your needs? SpeedWriter makes BASIC FLY! A must for the sophisticated programmer, yet easy enough for the novice. Reduces program size up to 50%, in addition to speeding up your program to run 10-30 times faster. Couldn't be simpler! (Disk) List $49.95. **Sale $39.95.** *Coupon $34.95.*

AdventureWriter — You think playing computer adventure games is exciting? How about the thrill of creating your own? It's all in this package. Your own heroes and villians, monsters and magicians, castles and coffins. You create complete adventure games without programming, which run on your disks. As many adventures as you can dream up - your treasures, your own fiendish riddles, and your treache rous plots for your friends to figure out! You control locations, treasures, hazards, characters, all descriptions and the plot. Computer adventures are one of the fastest growing hobbies around. Send them in for evaluation, we could market them for you. * graphics included in Commodore. **Com-64 Version** (Disk) List $49.95. **Sale $39.95.** *Coupon $34.95.* **Atari Version** (Disk) List $39.95. **Sale $29.95.**



Dialog — You can create all the EDUCATIONAL programs you can imagine without programming with the DIALOG program. Dialog is a simple on-screen process which allows you to create the first time you try, programs to educate children, inform adults, or just to entertain. Everything from simple tests to training manuals to complete novels. You control all the text, questions and answers. Any presentation is made more alive with the human give and take of real questions and answers. The Dialog system writes all the computer code onto your disk -to distribute any way to like. Graphics included. (Disk) List $49.95. **Sale $39.95.** *Coupon $34.95*

ELF System — Finally, a simple way to create calculation/print programs. These are NON-FILE programs which let you: (A) Collect information; (B) Number crunch with all the conditional calculations you need; (C) Display the results to the screen or printer any way you'd like. It's like a spreadsheet without rows and columns an a super calculating word processor without the hassle. (Disk) List $39.95. **Sale $29.95.** *Coupon $24.95.*

Programming Made Easy

Add $3.00 for shipping, handling and insurance. Illinois residents please add 6% tax. Add $6.00 for CANADA, PUERTO RICO, HAWAII, ALASKA, APO-FPO orders. Canadian orders must be in U.S. dollars. WE DO NOT EXPORT TO OTHER COUNTRIES, EXCEPT CANADA.
Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail!
VISA — MASTER CARD — C.O.D.
No C.O.D. to Canada, APO-FPO.

PROTECTO
We Love Our Customers
22292 N. Pepper Rd., Barrington, Illinois 60010
312/382-5244 to order

Reader Service No. 208

American Educational

$24.95 Computer $24.95

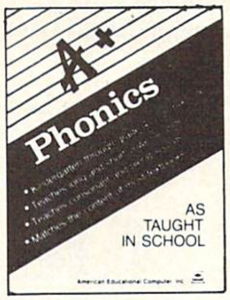
• Learn at home • Easy To Use • Colorful Graphics • Motivating and Fun • Teacher Designed

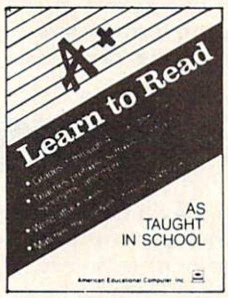
All programs were written by teachers and have been classroom validated. These programs teach in a standard classroom format with rewards as the user gets through the various lessons.

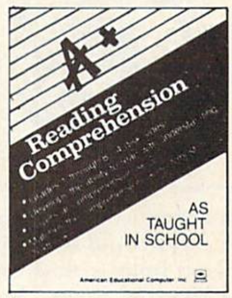
Atari & Commodore 64

EasyReader Series

There are three programs in this series designed and tested by teachers from the appropriate grade levels for accurate content. Each disk keeps track of the students name, ensuring that the user completes one lesson before starting a new one. Animation, Graphics and sound are used in a successful way that keeps the child's interest as he learns each new skill. Correct answers are rewarded and wrong answers are corrected. Very easy to use, with most instructions appearing on the screen when needed.





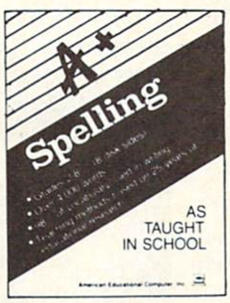


Phonics. A program designed to teach beginning phonics: the sound of vowels and consonants. Helps child to develop skills for successful reading. The material covered follows the same guidelines of most textbooks. 8 disk sides; for Kindergarten through grade 4. (Disk)
List $39.95. **Sale $24.95.**

Learn To Read. Provides work with structural word skills — contractions, prefixes, suffixes, synonyms and more. Prepares child for material that will be covered in the classroom as well as building on traditional school curriculum. Grades 1-4. (Disk) List $39.95. **Sale $24.95.**

Reading Comprehension. Helps child to develop reading skills with an accent on understanding the material presented. Teaches cause and effect, similarities and differences, predicting outcomes, main ideas and more. Grades 1-8. (Disk)
List $39.95. **Sale $24.95.**

Atari & Commodore 64



(Disk)

AEC Spelling — A series of seven educational software programs for grades 2 through 8. It teaches the spelling of 4,000 words most commonly used in writing (98% of most people's writing vocabulary). The word lists used are the result of over 25 years of research into the writing/spelling needs of children and adults. Each grade level consists of one two-sided disk with word lists on one side and study activities on the other. The activities are designed to develop mastery in spelling utilizing a Test/Teach/Test approach. A pre-test is used to find out which words a user cannot spell correctly. These are recorded on a "Words To Study" list. The activities for learning these words are provided. Finally, a post-test is taken by the learner to measure progress. The approach is simple and rewards are built in. (Grades 1-8) List $39.95. **Special Sale — 8 programs/4 disks $24.95.**



(Disk)

Add $3.00 for shipping, handling and insurance. Illinois residents please add 6% tax. Add $6.00 for CANADA, PUERTO RICO, HAWAII, ALASKA, APO-FPO orders. Canadian orders must be in U.S. dollars. WE DO NOT EXPORT TO OTHER COUNTRIES, EXCEPT CANADA.

Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail!

VISA — MASTER CARD — C.O.D.

No C.O.D. to Canada, APO-FPO.

PROTECTO

We Love Our Customers

22292 N. Pepper Rd., Barrington, Illinois 60010

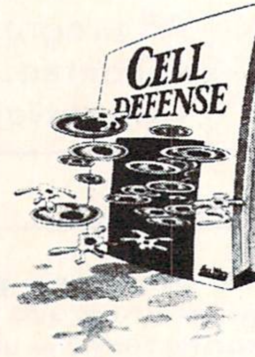
312/382-5244 to order

Reader Service No. 208

Fantastic Sale!

HES Special Programs For Your Computer

While They Last!! **$9.95** **Call Today!!** **$9.95** **First Come Basis!!**



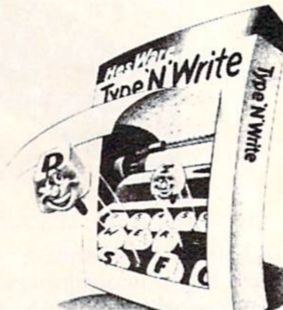
Cell Defense — Quick. Deadly viruses are rapidly multiplying. If you don't act fast, they'll infiltrate your entire bloodstream. And in seconds it'll be over. So blast the invaders with interferon. And annihilate them with macrophages. With Cell Defense, you control your own immune system. You'll discover basic concepts in biology. And explore exciting scientific strategies. While at the same time, leading an attack against a host of deadly viruses. So get yourself the science simulation game Cell Defense. Your life may depend on it. (Disk) List $34.95. **Early Christmas Sale $9.95.**



Commodore 64

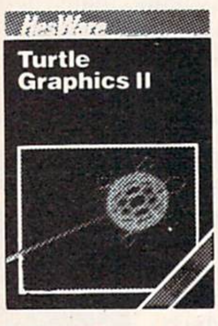
Super Zaxxon — Use the skills and tactics you learned in Zaxxon to master the new patterns and dangers in Super Zaxxon. Floating foretresses but with deadlier weapons and tighter passageways. 3-dimensional perspective but with even more realistic graphics. 3-dimensional flight but with a perilous tunnel requiring even more precise control. And a fire-breathing dragon that makes Zaxxon's robot look like a wimp. (Disk) List $29.95. **Early Christmas Sale $9.95.**

Early



Typing Tutor — Give your children an education they'll never get in school. Just think how much a child could learn if they had their very own private teacher at home. That kind of specialized help is now available through educational software programs from HesWare. There's Type 'N' Write, for the Commodore 64, that introduces your children to keyboard skills. Touch typing. And even word processing, for creating and editing up to eight pages. (Disk) List $29.95. **Early Christmas Sale $9.95.**

Christmas



Turtle Graphics II (LOGO) — An exciting and fun way to discover computer programming concepts. Turtle Graphics II is an easy-to-learn computer language that enables the beginning programmer to master the computer in just a few minutes. By telling an imaginary turtle how to roam about on the screen, the Turtle Graphics II programmer can create colorful and artistic pictures. As one becomes more and more adept at teaching the turtle to paint, he or she is painlessly learning all of the basics of computer programming. This not only develops computer awareness but enriches one's math, logic, communication and artistic skills as well. List $29.95. **Early Christmas Sale $9.95.**

Sale



POOL CHALLENGE — 6 Games in One • Endorsed by Minnesota Fats • Great Action

Now you can play pool on your Commodore-64. Sharpen your aiming accuracy and shooting finesse. Using only a joystick and a few keystrokes you choose where to aim and how much power to use. Fantastic action. (Cartridge). List $29.95. **Early Christmas Sale $9.95.**

Add $3.00 for shipping, handling and insurance. Illinois residents please add 6% tax. Add $6.00 for CANADA, PUERTO RICO, HAWAII, ALASKA, APO-FPO orders. Canadian orders must be in U.S. dollars. WE DO NOT EXPORT TO OTHER COUNTRIES, EXCEPT CANADA.

Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail!

VISA — MASTER CARD — C.O.D.

No C.O.D. to Canada, APO-FPO.

PROTECTO

We Love Our Customers

22292 N. Pepper Rd., Barrington, Illinois 60010

312/382-5244 to order

Reader Service No. 208



Aquarian Software

P.O. Box 184
Portland, OR 97222

To Order, Call: (503) 654-2641

VISA and MasterCard Accepted

Add $3.00 Shipping and Handling Per Order

Dealer Inquiries Welcomed!

Modem Master

The Fastest Commodore BBS Available!

- Works with 1541 or MSD Dual Drive
- 300/1200 Baud Operation
- New Punter File Transfer Protocol
- Fully Menu Driven
- Over 25 Commands
- Public Message Base
- Private E-Mail Base
- E-Mail Check at Sign-On
- Sub-Directories for File Transfer
- 250 User Capacity
- Accurate Clock/Calendar
- Printer Output
- Information Files
- "Old" E-Mail Deleted After One Week
- User Survey/Poll
- Set Up in Only 10 Minutes!

Only $29.95

Omiterm Terminal

- Fully supports the new 1660 Modem 300!
- Punter protocol — upload & download — 300/1200 baud
- Ten programmable function keys
- 15 number phone directory
- 20k receive buffer
- Tone or pulse dialing
- Auto dial/re-dial
- Half/full duplex

$19.95

COMPUTER TALK

Guides the novice through the maze of buzz words and mysteries of telecommunications. Explains commonly used terms, advises on selection of both hardware and software. Introduces you to both BBS's and national networks.

Only $8.00

Includes PlayNET demo disk with two FREE hours on-line!

Disk-Lock V2.1

At Last! A professional copy protection system is available to YOU. Now you can protect your programs from most copy programs!

- Disk-Lock is Designed to Prevent Full Disk Copies
- Stops Over 90% of All Copiers
- Offers Multiple Protection Schemes
- Place Protection on Any Track From 1 to 40!
- Fully Menu Driven

Only $39.95

The Cataloger

The Ultimate Disk Cataloging System for the 64!

Will find any program instantly. Tells exactly which disk number the program is on, the disk name, disk ID, front or back of disk, category (games, utility, music, etc.), if the disk is protected (if so, what errors), the date the program was entered, the block count, and any comments concerning that program. Sort and print a list by any of 12 different fields.

Features of The Cataloger V3.5A Include:

- ★ Easy loading of information. Loads directly from the disk itself.
- ★ Ability to change name of entry.
- ★ Ability to change any information instantly — uses relative files exclusively.
- ★ Search, Sort and Print by any of 12 fields.
- ★ Capacity for 1100 different programs (or disks) per data disk.
- ★ FAST — all machine language.
- ★ Menu driven — very easy to use.
- ★ Works with ALL printers.
- ★ Works with one or two drives.
- ★ Allows duplicate ID's.
- ★ Allows selection of items to be entered.
- ★ Unlimited category names.
- ★ FREE updates — new features added periodically.

Only $24.95

A Survival Guide To The 1541 Disk Drive

If you are bewildered by the 1541 User's Manual, then help is here at last! In-depth explanations of your disk drive and how to use it, clearly explained in simple, easy-to-understand terms! A real help!

Order Today! Only $5.00

TURBO CALC/64

A Great Beginner's Spreadsheet At an UNBELIEVABLE Price!!

- 100% Memory Resident
- 100% Machine Code and P-Code
- 100% Menu Driven
- Over 13 K RAM for Data
- Large Matrix, 100 Rows by 26 Columns
- On-Line Help Screens
- On-Screen Menu at All Times
- Full Printer Support
- Built-in Functions (SUM, AVG, MIN, MAX)

At last, a powerful, yet affordable spreadsheet!

ONLY $17.95

B.E.S.T.

At last! A truly professional accounting system for the Commodore 64 that is easy to use!

General Ledger	**$69.95**
Accounts Payable	**$59.95**
Accounts Receivable	**$59.95**
Inventory	**$69.95**

Big Computer Power At Small Computer Prices!!

Call For Full Details

Thriller Series

All six thrilling adventures now together on one disk! Plus — as a bonus — Three Hours To Live! These are full-length machine language text adventures that will take you weeks to complete. They feature full-sentence input (not just two words!) with color and sound.

Includes:

- Night of the Walking Dead
- Son of Ali Baba
- Perils of Darkest Africa
- Frankenstein's Legacy
- Revenge of The Moon Goddess
- Shipwrecked!
- Plus Three Hours to Live

Only $24.95
For All SEVEN!

Nova Decoder 64

Version 3.0

A 6502/6510 Disassembler
Designed for the Commodore User

A MUST for anyone just learning Machine Language.
A TREAT for those examining the operation of the C-64.
A BLESSING to machine language programmers trying to de-bug their programs.

- ★ Choose Screen or Printer Options
- ★ Addresses Printed in Either Hex or Decimal
- ★ Complete Calculations of Branch Commands
- ★ Brief Explanations of Each Command Encountered
- ★ Generates Tables of Zero Page Addresses Used, Branch Addresses, All Jump Routines, and More . . .

Only $19.95

I.E.A. Instant Editor Assembler

for the Commodore 64

- ★ Written 100% in machine language
- ★ Assembles 17K source code in 5 seconds!
- ★ Co-resident editor assembler monitor
- ★ Compatible with HES MON
- ★ Slow-motion code analyzer
- ★ Technical assistance number

"I would have to recommend the IEA Instant Editor Assembler, the BEST VALUE-FOR-PRICE development package on the market!"
Jim Gracely
Technical Editor
Commodore Microcomputer Magazine

"IEA is a nice, simple assembler, good for beginners at an UNBELIEVABLE PRICE!"
Jim Butterfield

"...Should be called the POOR MAN'S MERLIN — a great assembler, and you can't beat the price."
William B. Sanders
Author of Assembly Language For Kids

Now Only $17.95

SHADEYDUMP

for the C-64 and Gemini Printer

BY MICHAEL SCHELL

Some of the most popular software packages for the Commodore 64 are graphics programs. Using these, one can produce remarkable color pictures on the screen. Unfortunately, few screen dumps can account for color. By printing a different 4x4 dot pattern for each of the 16 colors, *ShadeyDump* produces a gray scale printout of any bitmapped hi-res (or multicolored) screen consisting of over a million dots.

Written in 100% machine language to be used with the Gemini 10X double-density, double-speed graphics mode, *ShadeyDump* can print either standard bit-mapped screens or multicolor bit-mapped screens. *ShadeyDump* can print a mirror image of the screen. *ShadeyDump* can be used with any hardware printer interface, or with a user port software driven interface. *ShadeyDump* even includes built-in BLOCKLOAD and BLOCKSAVE commands to simplify loading of hi-res screens.

USING *SHADEYDUMP*

When loaded, *ShadeyDump* assumes several things. The first is that your picture is a standard hi-res image located at 24576 and the color information is stored at 23552. This format is compatible with pictures created with the *DOODLE!* graphics package. Second, *ShadeyDump* assumes that you are using a printer interface which responds to device number 4, and in transparent mode with secondary address 5. If either of these assumptions are wrong you will need to change some of *ShadeyDump*'s pointers.

To *ShadeyDump* a *DOODLE!*-compatible picture, sim-

SHADEYDUMP SOURCE CODE LISTING

```
LOC  LINE
0000 ;*********************************
0000 ;*                               *
0000 ;*          SHADEYDUMP           *
0000 ;*         VERSION  2.0          *
0000 ;*                               *
0000 ;*  (A GRAY SCALE SCREEN DUMP)   *
0000 ;*                               *
0000 ;* (C) 1985 BY MICHAEL SCHELL    *
0000 ;*                               *
0000 ;*********************************
0000 ;*
0000 *       = $C000
C000 ;*********************************
C000 ;*                               *
C000 ;*           EQUATES             *
C000 ;*                               *
C000 ;*********************************
C000 ;*
C000 BASE   = $57
C000 COLPTR = $59
C000 COLUMN = $5B
C000 DOTS   = $5C
C000 COLHLF = $5D
C000 DBLSTR = $5E
C000 ROW    = $5F
C000 CBYTE  = $60
C000 FAC1   = $61
C000 FAC2   = $69
C000 COLOR  = $FB
C000 ;*
C000 CIAIOR = $DD01
C000 CIADDR = $DD03
C000 CIAICR = $DD0D
C000 ;*
C000 ICHROT = $326
C000 SETLFS = $FFBA
C000 SETNAM = $FFBD
C000 OPEN   = $FFC0
C000 KCLOSE = $FFC3
C000 CHKOUT = $FFC9
C000 CLRCHN = $FFCC
C000 CHROUT = $FFD2
C000 ;*********************************
C000 SCRNDP JSR CONECT
C003        LDX #0
C005 LOOP1  STX COLUMN
C007        LDX #1
C009 LOOP2  STX COLHLF
C00B        JSR NXTCOL
C00E        LDX #0
C010 LOOP3  STX DBLSTR
C012        JSR INTLIN
C015        LDA RVSFLG
C018        BEQ RVS1
C01A        LDX #24
C01C        .BYTE $2C      ;BIT
C01D RVS1   LDX #0
C01F LOOP4  STX ROW
C021        JSR SETCOL
C024        LDA RVSFLG
C027        BEQ RVS2
C029        LDX #7
C02B        .BYTE $2C      ;BIT
C02C RVS2   LDX #0
C02E LOOP5  STX CBYTE
C030        JSR PRTCHR
C033        LDA RVSFLG
C036        BEQ RVS3
C038        LDX CBYTE
C03A        DEX
C03B        BPL LOOP5
C03D        LDX ROW
C03F        DEX
C040        BPL LOOP4
C042        BMI SKIP1
C044 RVS3   LDX CBYTE
C046        INX
C047        CPX #8
C049        BNE LOOP5
C04B        LDX ROW
C04D        INX
C04E        CPX #25
C050        BNE LOOP4
C052 SKIP1  LDX DBLSTR
C054        INX
C055        CPX #2
C057        BNE LOOP3
C059        LDX COLHLF
C05B        DEX
C05C        BPL LOOP2
C05E        LDX COLUMN
C060        INX
C061        CPX #40
C063        BNE LOOP1
C065        JMP CLOSE
C068 ;*
C068 ;*********************************
C068 ;*                               *
C068 ;*       PRINTER ROUTINES        *
C068 ;*                               *
C068 ;*********************************
C068 ;*
C068 CONECT LDA INTMOD
C06B        BEQ SERIAL
C06D PARALL LDA #$FF
C06F        STA CIADDR
C072        LDA ICHROT
C075        STA VECTOR
C078        LDA ICHROT+1
C07B        STA VECTOR+1
C07E        LDA #<PRTOUT
C080        STA ICHROT
C083        LDA #>PRTOUT
C085        STA ICHROT+1
C088        BNE SETLF
C08A ;*********************************
C08A SERIAL LDA #4
C08C        LDX DEVNUM
C08F        LDY SECADD
C092        JSR SETLFS
C095        LDA #0
C097        JSR SETNAM
C09A        JSR OPEN
C09D        LDX #4
C09F        JSR CHKOUT
C0A2 ;* * * * * * * * * * * * * * * * *
C0A2 SETLF  JSR ESCAPE
C0A5        LDA #51
C0A7        JSR CHROUT
C0AA        LDA #1
C0AC ;* * * * * * * * * * * * * * * * *
C0AC CRRT1  JSR CHROUT
C0AF ;* * * * * * * * * * * * * * * * *
C0AF CRRT   LDA #13
C0B1        JSR CHROUT
C0B4        LDA #10
C0B6        JMP CHROUT
C0B9 ;*********************************
C0B9 ESCAPE LDA #27
C0BB        JMP CHROUT
C0BE ;*********************************
C0BE PRTOUT PHA
C0BF        STA CIAIOR
C0C2        LDA #$10
C0C4 ACKCHR BIT CIAICR
C0C7        BEQ ACKCHR
C0C9        PLA
C0CA        RTS
C0CB ;*********************************
C0CB NXTCOL JSR ESCAPE
C0CE        LDA #74
C0D0        JSR CHROUT
C0D3        LDA #13
C0D5        JMP CRRT1
C0D8 ;*********************************
C0D8 INTLIN JSR CRRT
C0DB        JSR ESCAPE
```

ply load the picture using:

```
LOAD "(NAME)",8,1
```

After loading your picture run *ShadeyDump* with:

```
SYS 49152
```

If you wish to print a hi-res image located somewhere else in memory you will have to change two pointers. You can POKE 49704 and 49705 with the proper address of the bit-map in standard low-byte/high-byte format and change the pointer to the video matrix at 49706 and 49707, but *ShadeyDump* can do this for you. Simply use the following *ShadeyDump* SYS command:

```
SYS 49936, (ADDRESS OF BIT MAP),(ADDRESS
OF VIDEO MATRIX)
```

Example: to *ShadeyDump* a *DOODLE!* image enter this line:

```
SYS 49936,24576,23552
```

ShadeyDump will automatically POKE all the proper locations to print your picture.

Printing a multicolor picture requires changing four pointers, the pointer to the bit map at 49704, the pointer to the video matrix at 49706, the pointer to color memory at 49708, and the pointer to the screen color at 49710, and toggling setting one flag at 49703 to zero. You can use POKEs to set these pointers, or you can again use a *ShadeyDump* SYS command:

```
SYS 49960, (ADDRESS OF BIT MAP),(VIDEO M
ATRIX),(COLOR MEMORY),(SCREEN COLOR)
```

Example: to *ShadeyDump* a *Koala* image enter this line:

```
SYS 49960,24576,32576,33576,34576
```

Again *ShadeyDump* will automatically POKE all the proper locations.

Most users of *ShadeyDump* will use a serial port interface such as those made by Cardco. *ShadeyDump* defaults to this type of interface with a device number of 4 and secondary address of 5. To change the device number POKE 49716 with the proper number, and to change the secondary address POKE 49717 with the proper value.

Some users will prefer to use a parallel interface connected through the user port. *ShadeyDump* has a built-in driver routine to use this type of interface. To access this feature POKE 49713 with a one to toggle the interface flag.

ADVANCED *SHADEYDUMP* TOPICS

If the standard version of *ShadeyDump* defaults to values that are not optimal for your use, you can easily save a custom version of *ShadeyDump* using *ShadeyDump*'s built-in BLOCKSAVE SYS command. After changing *ShadeyDump*'s pointers and flags, save your new version of *ShadeyDump* with:

```
C0DE        LDA #121
C0E0        JSR CHROUT
C0E3        LDA #32
C0E5        JSR CHROUT
C0E8        LDA #3
C0EA        JMP CHROUT
C0ED ;*********************************
C0ED CLOSE  LDA INTMOD
C0F0        BEQ SCLOSE
C0F2 PCLOSE LDA VECTOR
C0F5        STA ICHROT
C0F8        LDA VECTOR+1
C0FB        STA ICHROT+1
C0FE        RTS
C0FF SCLOSE JSR CLRCHN
C102        LDA #4
C104        JMP KCLOSE
C107 ;*
C107 ;*********************************
C107 ;*                               *
C107 ;*      MATH ROUTINES            *
C107 ;*                               *
C107 ;*********************************
C107        .MAC TIMES
C107        LDA ?1
C107        STA FAC2+2
C107        LDA #?2
C107        STA FAC2+3
C107        JSR MULTPY
C107        .MND
C107 ;********************************
C107 MULTPY LDA #0
C109        STA FAC2
C10B        STA FAC2+1
C10D        LDX #8
C10F SHIFT  ASL A
C110        ROL FAC2+1
C112        ASL FAC2+2
C114        BCC CHCNT
C116        CLC
C117        ADC FAC2+3
C119        BCC CHCNT
C11B        INC FAC2+1
C11D CHCNT  DEX
C11E        BNE SHIFT
C120        STA FAC2
C122        RTS
C123 ;*********************************
C123 SETCOL TIMES ROW,40
C12E        LDA COLUMN
C130        CLC
C131        ADC FAC2
C133        STA FAC2
C135        BCC SKIP2
C137        INC FAC2+1
C139 SKIP2  LDA FAC2
C13B        STA BASE
C13D        LDA FAC2+1
C13F        STA BASE+1
C141        LDA SCREEN
C144        CLC
C145        ADC FAC2
C147        STA FAC2
C149        STA COLPTR
C14B        LDA SCREEN+1
C14E        ADC FAC2+1
C150        STA FAC2+1
C152        STA COLPTR+1
C154        LDY #0
C156        LDA (COLPTR),Y
C158        AND #15
C15A        STA COLOR+1
C15C        LDA (COLPTR),Y
C15E        LSR A
C15F        LSR A
C160        LSR A
C161        LSR A
C162        STA COLOR+2
C164        LDA BASE+1
C166        STA FAC2+1
C168        LDA BASE
C16A        STA FAC2
C16C        CLC
C16D        ADC COLMEM
C170        STA FAC2
C172        LDA BASE+1
C174        ADC COLMEM+1
C177        STA FAC2+1
C179        LDY #00
C17B        LDA (FAC2),Y
C17D        AND #15
C17F        STA COLOR
C181        LDA BCKGRN
C184        STA FAC2
C186        LDA BCKGRN+1
C189        STA FAC2+1
C18B        LDA (FAC2),Y
C18D        AND #15
C18F        STA COLOR+4
C191        LDA BASE+1
C193        STA FAC2+1
C195        LDA BASE
C197        STA FAC2
C199        LDX #03
C19B LOOP8  ASL FAC2
C19D        ROL FAC2+1
C19F        DEX
C1A0        BNE LOOP8
C1A2        LDA HIRES
C1A5        CLC
C1A6        ADC FAC2
C1A8        STA FAC2
C1AA        LDA HIRES+1
C1AD        ADC FAC2+1
C1AF        STA FAC2+1
C1B1        RTS
C1B2 ;*********************************
C1B2 PRTCHR LDY CBYTE
C1B4        LDA (FAC2),Y
C1B6        STA FAC2+2
C1B8        LDA COLHLF
C1BA        BNE SKIP4
C1BC        LDX #4
C1BE LOOPA  ASL FAC2+2
C1C0        DEX
C1C1        BNE LOOPA
C1C3 SKIP4  LDY #3
C1C5 LOOPB  STY DOTS
C1C7        LDA MCMODE
C1CA        BEQ MCDMP
C1CC HIRDMP LDY #4
C1CE        LDA FAC2+2
C1D0        STA FAC2+3
C1D2 HLOOPC ASL FAC2+3
C1D4        LDA #00
C1D6        ROL A
C1D7        TAX
C1D8        INX
C1D9        BNE SKIP12
C1DB MCDMP  LDY #2
C1DD        LDA FAC2+2
C1DF        STA FAC2+3
C1E1 MLOOPC ASL FAC2+3
C1E3        LDA #00
C1E5        ROL A
C1E6        ASL FAC2+3
C1E8        ROL A
C1E9        EOR #3
C1EB        TAX
C1EC SKIP12 LDA COLOR,X
```

```
SYS 49876,49152,50010,"(NAME)",(DEV. NO.)
```

Using this feature it is possible to have a different version of *ShadeyDump* for every graphic package that you own.

The proper syntax for the BLOCKSAVE command is:

```
SYS 49876, (STARTING ADDRESS),(ENDING AD
DRESS+1),"(FILE NAME)",(DEV. NO.)
```

ShadeyDump also includes a BLOCKLOAD SYS command to allow you to load files to non-standard locations. The proper syntax is:

```
SYS 49909, (LOAD ADDRESS),"(NAME)",(DEV.
NO.)
```

ShadeyDump's dot matrix is stored in memory starting at 49734 and running to 49861. The dot matrix runs from lightest color to darkest. Each color is represented by eight bytes, four for the first strike and four for the second strike. *ShadeyDump* only uses the two least significant bits of each byte in printing the dot pattern. Using a monitor or simply POKEing values (0-3) into these locations you can change, and improve, the dot patterns *ShadeyDump* uses to print gray scales. You can save your new improved version of *ShadeyDump* with the BLOCKSAVE command.

Finally, *ShadeyDump* can produce mirror images of the pictures it dumps. To print a mirror image simply POKE 49712 with a zero. □

SEE PROGRAM LISTING ON PAGE 139

SHADEYDUMP MULTICOLOR FILE POINTERS By Morton Kevelson

ShadeyDump can generate a gray scale printout for any multicolor image, not just *Koala* files. Here is a list of file parameters for some popular multicolor drawing packages for the C-64. The table lists the parameters in the proper format for the *ShadeyDump* SYS command:

```
SYS 49960,BMP,VM,CM,SC
```

Where: BMP = bit map
VM = video matrix
CM = color memory
SC = screen color

	NOTE	BMP	VM	CM	SC
Koala Printer	1	24576	32576	33576	34576
Peripheral Vision	2	24576	23552	32768	24575
Paint Magic	3	16384	24576	25576	24384
Flying Colors		8192	7168	6144	8191
Blazing Paddles	4	24576	32768	33792	32640
Micro Illustrator	5	8384	6384	7384	6383

Notes:

All files should be LOADed using a nonrelocating LOAD "filename",8,1 except as noted below.

1. The only thing unusual about *Koala* files is the structure of the file name. The first character is always shifted reverse "A" (CHR$(129)). The file name is always exactly 15 characters long. If the text of the name is not long enough it is padded with spaces (CHR$(32). If you are having trouble LOADing a *Koala* file then try using the DOS wild card characters; for example, "?PIC A NAME*".

2. *Peripheral Vision* stores its multicolor images in two files, one for the bit map and video matrix and the other for color memory. The first character of the bit map file name is the British

```
C1EE        TAX
C1EF        LDA SHADE,X
C1F2        ASL A
C1F3        CLC
C1F4        ADC DBLSTR
C1F6        ASL A
C1F7        ASL A
C1F8        ADC DOTS
C1FA        TAX ;X=((SHADE*2)+DBLSTR)*4
C1FB        LDA MCMODE
C1FE        BNE SKIP13
C200        LDA MATRIX,X
C203        LSR A
C204        ROL FAC2+4
C206        LSR A
C207        ROL FAC2+4
C209 SKIP13 LDA MATRIX,X
C20C        LSR A
C20D        ROL FAC2+4
C20F        LSR A
C210        ROL FAC2+4
C212        DEY
C213        BEQ SKIP14
C215        LDA MCMODE
C218        BEQ MLOOPC
C21A        BNE HLOOPC
C21C SKIP14 LDA FAC2+4
C21E        JSR CHROUT
C221        LDY DOTS
C223        DEY
C224        BPL LOOPB
C226        RTS
C227 ;*******************************
C227 MCMODE .BYTE 255
C228 SCREEN .WORD $5C00
C22A HIRES  .WORD $6000
C22C COLMEM .WORD $8000
C22E BCKGRN .WORD $D021
C230 RVSFLG .BYTE 255
C231 INTMOD .BYTE 0
C232 VECTOR .BYTE 0,0
C233
C234 DEVNUM .BYTE 4
C235 SECADD .BYTE 5
C236 ;*******************************
C236 SHADE  .BYTE 15,0,11,4,10,6,14,1
C237
C238
C239
C23A
C23B
C23C
C23D
C23E        .BYTE 9,13,5,12,7,2,8,3
C23F
C240
C241
C242
C243
C244
C245
C246 ;*******************************
C246 MATRIX .BYTE 0,0,0,0,0,0,0,0
C247
C248
C249
C24A
C24B
C24C
C24D
C24E        .BYTE 0,0,1,0,0,2,0,0
C24F
C250
C251
C252
C253
C254
C255
C256        .BYTE 1,0,0,0,0,2,0,0
C257
C258
C259
C25A
C25B
C25C
C25D
C25E        .BYTE 0,1,2,0,1,2,0,1
C25F
C260
C261
C262
C263
C264
C265
C266        .BYTE 2,1,0,2,0,2,1,0
C267
C268
C269
C26A
C26B
C26C
C26D
C26E        .BYTE 0,3,0,0,3,0,0,0
C26F
C270
C271
C272
C273
C274
C275
C276        .BYTE 3,0,0,0,0,3,0,0
C277
C278
C279
C27A
C27B
C27C
C27D
C27E        .BYTE 1,0,2,0,2,0,2,0
C27F
C280
C281
C282
C283
C284
C285
C286        .BYTE 0,2,0,2,0,2,0,1
C287
C288
C289
C28A
C28B
C28C
C28D
C28E        .BYTE 1,0,2,1,2,1,0,2
C28F
C290
C291
C292
C293
C294
C295
C296        .BYTE 1,2,0,1,2,0,1,2
C297
C298
C299
C29A
C29B
C29C
C29D
```

Pound (£) symbol (CHR$(92)). The last four characters are ".BMP". The rest of the file name can contain up to 11 user assigned characters. The last four characters of the color memory file are ".CMM". The remaining characters of the file name are the same 11 user assigned characters as above. The color memory file must not be LOADed using a nonrelocating LOAD. It normally occupies the same memory as *ShadeyDump.* Instead use *ShadeyDump's* BLOCKLOAD SYS command to relocate the color memory to a safe location:

```
SYS 49909,32768,"FILENAME.CMM",8
```

Peripheral Vision does not save the screen background color byte. It is always assumed to be white. Before executing *ShadeyDump* you must POKE 24575,1 in order to assign the proper screen background color.

3. *Paint Magic* files can be LOADed in two ways. The first, as a BASIC LOAD, allows the *Paint Magic* file to be viewed by simply RUNning it. The other is a nonrelocating LOAD. Use only the nonrelocating LOAD "file name",8,1 for *Paint Magic* with *ShadeyDump. Paint Magic* saves only a single byte which is used to fill the entire 1000 bytes of color memory. The color value for color memory is stored in location 24387. Since *ShadeyDump* expects a complete 1,000 byte color memory for a multicolor file, you will have to provide it. After LOADing the paint magic file and before executing *ShadeyDump* simply RUN the following brief program.

```
10 A=PEEK(24387)
20 FOR I=25576 TO 26575
30 POKE I,A
40 NEXT I
```

This will set up the contents of color memory.

4. Most of the *Blazing Paddles* multicolor image file is located in the RAM hidden beneath the BASIC ROM. The video matrix and color memory information LOADs into the $C000 block which is occupied by *ShadeyDump.* Use the *ShadeyDump* BLOCKLOAD SYS command to relocate the *Blazing Paddles* file to 24576 by entering

```
SYS 49909,24576,"FILE NAME",8
```

The original *Blazing Paddles* file parameters are:

40960-48959 Bit Map
49023 Border Color (low nybble)
49024 Background Color (low nybble)
49152-50151 Video Matrix
50176-51175 Color Memory

Suncom's Animation Station uses a customized version of the *Blazing Paddles* software. Multicolor image files created with the Animation Station should be treated exactly as *Blazing Paddles* files.

5. The *Micro-Illustrator* multicolor files are brought in with a nonrelocating LOAD. The screen background color is assumed to be white. You will have to POKE 6383,1 before executing *ShadeyDump.*

Although both *Peripheral Vision* and *Blazing Paddles* have built-in gray scale printer dumps, you will still find *ShadeyDump* useful. As a stand-alone routine it may be readily incorporated into your own programs. Having access to the source code will allow you to customize *ShadeyDump* to your own needs. Most notably the gray scale can be adjusted to accommodate varying image densities. *ShadeyDump* images are proportioned very close to the original screen dimensionis and slightly different from those produced by *Blazing Paddles*. This could be desirable in many situations. □

```
C29E            .BYTE 3,0,0,0,0,0,3,0
C29F
C2A0
C2A1
C2A2
C2A3
C2A4
C2A5
C2A6            .BYTE 0,3,0,0,0,0,0,3
C2A7
C2A8
C2A9
C2AA
C2AB
C2AC
C2AD
C2AE            .BYTE 2,1,2,1,0,0,0,0
C2AF
C2B0
C2B1
C2B2
C2B3
C2B4
C2B5
C2B6            .BYTE 3,3,3,3,0,0,0,0
C2B7
C2B8
C2B9
C2BA
C2BB
C2BC
C2BD
C2BE            .BYTE 3,3,3,3,3,3,3,3
C2BF
C2C0
C2C1
C2C2
C2C3
C2C4
C2C5
C2C6  ;*******************************
C2C6  ;*
C2C6  ;*******************************
C2C6  ;*                             *
C2C6  ;*      BSAVE & BLOAD          *
C2C6  ;*                             *
C2C6  ;*******************************
C2C6  ;*
C2C6  ;** BSAVE EQUATES **
C2C6  ;*
```

```
C2C6  FRMNUM = $AD8A
C2C6  GETADR = $B7F7
C2C6  CHKCOM = $AEFD
C2C6  BSSAVE = $E15F
C2C6  SETPAR = $E1D4
C2C6  BSLOAD = $E175
C2C6  ;*
C2C6  LINNUM = $14
C2C6  ;*
C2C6  GETPAR JSR CHKCOM
C2C9         JSR FRMNUM
C2CC         JSR GETADR
C2CF         LDA LINNUM
C2D1         LDX LINNUM+1
C2D3         RTS
C2D4  ;*
C2D4  BSAVE  JSR GETPAR
C2D7         PHA
C2D8         TXA
C2D9         PHA
C2DA         JSR GETPAR
C2DD         PHA
C2DE         TXA
C2DF         PHA
C2E0         JSR CHKCOM
C2E3         JSR SETPAR
C2E6         PLA
C2E7         TAY
C2E8         PLA
C2E9         TAX
C2EA         PLA
C2EB         STA LINNUM+1
C2ED         PLA
C2EE         STA LINNUM
C2F0         LDA #LINNUM
C2F2         JMP BSSAVE
C2F5  ;*
C2F5  BLOAD  JSR GETPAR
C2F8         PHA
C2F9         TXA
C2FA         PHA
C2FB         JSR CHKCOM
C2FE         JSR SETPAR
C301         PLA
C302         TAY
C303         PLA
C304         TAX
C305         LDA #0
C307         STA $A
C309         JMP BSLOAD
```

```
0000  ;*******************************
0000  ;*                             *
0000  ;*    ADDENDUM: SHADEYDUMP     *
0000  ;*      VERSION 2.0            *
0000. ;*                             *
0000  ;* (C) 1985 MICHAEL SCHELL     *
0000  ;*                             *
0000  ;*******************************
0000  ;*
0000  *      = $C310
C310  ;*
C310  GETPAR = $C2C6
C310  MCMODE = $C227
C310  SCREEN = $C228
C310  HIRES  = $C22A
C310  COLMEM = $C22C
C310  BCKGRN = $C22E
C310  RVSFLG = $C230
C310  INTMOD = $C231
C310  DEVNUM = $C234
C310  SECADD = $C235
C310  ;*
C310  STROUT = $AB1E
C310  ;*
C310  SETHRS JSR GETPAR
C313         STA HIRES
C316         STX HIRES+1
C319         JSR GETPAR
C31C         STA SCREEN
C31F         STX SCREEN+1
C322         LDA #255
C324         STA MCMODE
C327         RTS
C328  ;*
C328  SETMC  JSR GETPAR
C32B         STA HIRES
C32E         STX HIRES+1
C331         JSR GETPAR
C334         STA SCREEN
C337         STX SCREEN+1
C33A         JSR GETPAR
C33D         STA COLMEM
C340         STX COLMEM+1
C343         JSR GETPAR
C346         STA BCKGRN
C349         STX BCKGRN+1
C34C         LDA #00
C34E         STA MCMODE
C351         RTS
C352  ;*
```

SHOTGUN

A Rapid-Fire Diskkeeping Utility for the 64

By R. Harold Droid

Disk housekeeping is a chore best reserved for rainy days when the basement floods, giving you an excuse to get out of mopping. It's not that a clean, neat, orderly file box repulses me; it's the aggravation it takes to get there. You've probably amassed a collection of neat little programs that do tidy little organizing jobs on disks, like copying files, printing out master directories, renaming and locking files. But many neat and tidy little programs don't make file box rejuvenation a neat and tidy little job. I don't really need any more practice typing "LOAD" and "RUN."

Obviously what I need is a user-definable, menu-driven, auto-loading and running, multi-BASIC program-booting piece of weaponry which is always available so I can get a new program into memory with a flick of my itchy trigger finger. I call it *Shotgun.* Although it is designed with diskkeeping in mind, it's going to be useful any time you have a collection of programs with complementary functions that you have to continually interchange to get your job done.

Using *Shotgun* is as easy as LOADing it with an ",8,1" appended and typing NEW and SYS51200. Press the SHIFT and RESTORE keys simultaneously to see your menu. From the menu, you can load and run one of up to 26 files, or press the space bar to run the currently resident BASIC program. There are two other functions available. Space/RESTORE instantly reruns the BASIC program in memory (even while it's still running) and Commodore/RESTORE does a BASIC warm start which leaves *Shotgun* available. RUN-STOP/RESTORE disables *Shotgun.* You can get it back with SYS51200.

Shotgun is a customized menu program which you create by specifying which files you wish to include. To get a *Shotgun* for a particular disk, use the program *Shotgun Shell* and *Shotgun Loader.* The loader program loads the shell into memory and reads a disk directory. Simply press 'Y' or 'N' when a file name is displayed. At the end of the directory, you are prompted for a file name for your new rapid-repeating, audo-load/run menu program, with the default name *Shotgun.* The loader program then saves it for you.

Shotgun runs off of the NMI interrupt of the 64. This interrupt is generated whenever the RESTORE key is pressed. When you SYS51200, the vector which points to the NMI interrupt routine is changed to point to *Shotgun.* Then every time you hit RESTORE, the SHIFT, Commodore, and space bar keys are checked. If one of these isn't pressed, control reverts to the normal NMI interrupt routine. At this point, if the RUN/STOP key isn't pressed, or if you don't have a cartridge plugged in, nothing happens and the computer continues on its merry way.

If one of the programs you load messes with the NMI vector, you're in for some strange results. None of the programs I'm using do. Note also that *Shotgun* sits in an area of memory that's heavily trafficked when BASIC needs machine language support. Most of the time programmers put such routines at 49152, which leaves eight free blocks before they overwrite *Shotgun.* Four blocks above are also open, so if you use the DOS wedge for doing disk chores, *Shotgun* is compatible. If you're as tired of practicing your spelling on "LOAD" and "RUN" as I am, this program should be compatible with you, too. □

SEE PROGRAM LISTING ON PAGE 132

READER SERVICE INDEX

Page No.	Company	Svc. No.
13	3M	200
14,15	Abacus Software	213
C-4	Access Software	184
63	Access Software	165
97	Ace Software	170
71	Alsoft	210
21	APCAD	171
92	Aquarian Software	214
72	BAMTEK	216
43	Big Bytes	244
111	Big L. Electronics	230
65	Broderbund Software	167
46	C.O.M.B. Co.	241
70	C.O.M.B. Co.	209
C-3	C.O.M.B. Co.	175
44	Central Point Software, Inc.	232
112	Cheatsheet Products, Inc.	173
45	Comal Users Group USA, Ltd.	–
24	Commander Electronic Systems	224
8	Commodore-Amiga Inc.	192
23	CompuServe	217
100,101	Computer Centers of America	231
10	Computer Management Corp.	178
99	Computel Publishing Society	243
9	Computel Publishing Society	207
41	Computel Publishing Society	179
107	Custom Programming Group	186
22	CVC Online	172
50	Electronic Arts	124
32	Full Circle Software, Inc.	187
15	Genesis Computer Corporation	201
116	H.B.H. Corp.	229
43	Hesware	236
16	Infocom Inc.	203
61	Integrated-Software Systems	174
57	Jason-Ranheim	205
55	Jason-Ranheim	185
12	Lifeware	198
52,53	Lyco	215
48,49	Mastertronic International Inc.	225
36	Megasoft Ltd.	219
17	Megasoft Ltd.	218
56	Micro-W Distributing, Inc.	206
105	Microcomscribe	189
42	Microlog	188
4	Microvations/Houseworks	234
30	Navarone Industries, Inc.	226
44	OWI Incorporated	190
5	PDS Sports	168
33	Peripheral Systems of America	228
102	Post Technologies	238
16	Powerline Software	204
12	Powerline Software	197
99	Professor Jones	235
74-91	Protecto Enterprizes	208
11	Proteus Electronics Inc.	194
26	Quantum Software	223
73	Quorum International Unltd.	242
106	Roger Wagner Publishing, Inc.	221
54	Sachs Enterprises	245
11	Sakata U.S.A. Corporation	193
58	Schnedler Systems	–
27	Skyles Electric Works	183
15	Spectron Instrument	202
64	Strategic Simulations, Inc.	166
C-2	SUBLogic Corporation	176
108	TPUG Inc.	162
73	T&D Subscription Software	240
38	Talktronics, Inc.	222
20	Tenex Computer Express	181
12	Timeworks, Inc.	199
12	Timeworks, Inc.	196
44	Tomy Corporation	191
69	Ultrabyte	211
98	Universal Marketing	239
6	Value-Soft	169
66,67	Viewtron	212
102	VMC Software	237
25	Wedgwood Rental	177
62	White House Computer	220
11	X-10 (USA) Inc.	195
44	Xetec Inc.	180

The publisher cannot assume responsibility for errors in the above listing.

SCUTTLEBUTT

Continued from page 16

view of the non-MIDI version of *The Music Shop* on page 65.)

Passport Designs, Inc., 625 Miramontes St. Half Moon Bay, CA 94019 (phone: 415-726-0280).

Two new C-64-compatible programs from Dr. T Music Software:

The *Echo Plus* MIDI delay program, in addition to performing the standard digital-delay functions, will make the C-64 "listen" to one MIDI keyboard and send MIDI commands on up to four channels to any keyboards connected to the computer's MIDI output. Effects that can be created include keyboard splitting, doubling, echoing, one-finger and arpeggiated chords, and infinite loops. The $90 program runs with a Sequential, Passport, Korg, Yamaha, or Music-Data MIDI interface.

The *CZ Patch Librarian* is for use in creating, editing, and storing patches on the Casio CZ-101, CZ-10000, and CZ-5000 synthesizers.

Dr. T Music Software, 190 Washington Ave., Brooklyn, NY 11205 (phone: 718-625-7283).

RolandCorp US has released the *MUSE* (MIDI Users Sequencer/Editor) software system for the C-64, requiring RolandCorp's MPU-401 MIDI Processing Unit and a MIDI instrument. It allows the C-64 to perform 8-track recording and overdubbing, time correction of any tracks, full editing by measure, and more.

RolandCorp US, 7200 Dominion Circle, Los Angeles, CA 90040-3647 (phone: 213-685-5141).

COMPUSERVE CHANGES

The financial, demographic, and editorial subject areas of the CompuServe Information Service, previously available only to Executive Information Service subscribers, are now accessible to all 225,000 subscribers through an Executive Service Option. Price of a CIS subscription kit will remain $39.95; a onetime fee of $10 ($5 for new customers) will be charged for the Executive Service Option (there will be a $10 monthly usage minimum).

The CompuServe Information Service has been further restructured to include "tour" and "find" options, article paging (fits articles to user's screen size), a more detailed, easier to use top menu, and a list of quick reference words.

CompuServe Incorporated, 5000 Arlington Centre Boulevard, P.O. Box 20212, Columbus, OH 43220 (phone: 614-457-8600).

PRICE REDUCTIONS

The price of the *Deluxe COMAL 2.0 Cartridge Pak* has been reduced almost $40 to $89.95. Included is the cartridge, a 320-page tutorial guide, and five demo disks.

COMAL Users Group, USA, Ltd., 6041 Monona Dr., Madison, WI 53716 (phone: 608-222-4432).

The price of the Printmaster/+G interface has been reduced to $99.95.

Omnitronix Incorporated, P.O. Box 434, Mercer Island, WA 98040 (phone: 206-236-2983).

FLOTSAM

As this is being written, I am in New York State for a brief visit. I picked up a copy of *Ahoy!* today to pass the evenings. I use a Commodore 64 at home in Sacramento.

I am writing to counter the view expressed by one John W. Carino in the July issue. There is nothing offensive or dangerous to children in the advertisements of CVC Online.

On the other hand, there is a great deal of danger in the idea that a magazine should limit its advertisements to suit prudes, moralists, and others who believe children should be raised in barrels and fed through the bunghole until they are old enough to have the Real World sprung on them in all its glory and anti-glory.

Mr. Carino claims he is not a prude or a moralist. Well, if it walks like a duck, quacks like a duck, and deposits fertilizer like a duck, then one is safe in assuming the creature *is* a duck. Only a prude, moralist, or —worse—would request you to remove an ad as inoffensive as CVC Online's ad. I say to Mr. Carino: Sex is here to stay. Do you want your children prepared to deal with the real world, or do you want them prepared to live in a fantasy world? If you answer is the latter, then growing up is going to be a very painful process when they get out from under your thumb.

—Donald L. Cline
Sacramento, CA

P.S.: If you *do not* cancel CVC's ad, i.e., if you do not knuckle under to prudes and moralists, my subscription order will be in the mail when I return home.

Let me begin by saying that I, too, have been an enthusiastic reader of your magazine from the beginning.

However, I must raise a serious objection to the opinion expressed by John W. Carino in his letter published in issue #19. He complained about the ad from CVC Online, calling it an "advertisement of that type," warning of decreased circulation, expressing concern for the effect on children, etc. Although Mr. Carino says he is "not a prude or moralist," I searched his letter in vain for any *other* grounds for his complaint.

Please don't misunderstand me. I do think an editor has an obligation to his readers concerning his advertisers. If, for example, you receive complaints regarding one of your accounts (that they misrepresent their product or service, that they overcharge, etc.), then by all means drop them. But to drop an advertisement simply because one reader finds it offensive is to set a dangerous precedent.

Although I have never personally used CVC Online, they seem to be a reputable company offering a legitimate service at a reasonable cost. Nor do I find the ad offensive. What I *do* find offensive is the concept of censorship, that someone I don't even know would presume to influence what ad I may see.

I sincerely hope that it will not be the editorial policy of *Ahoy!* to cave in to this kind of pressure. So far, you have an excellent publication. Keep it that way.

—Mike Nichols
Kansas City, MO

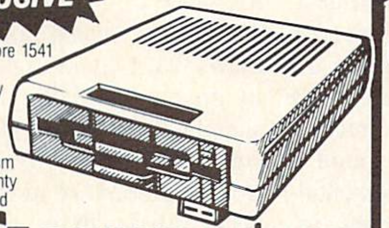
Thank you very much for your excellent article in the May '85 issue, *Install a Reset Switch on Your VIC/64* by Tom Benford. I followed the well-written instructions and the setup works fine.

I would like to offer a couple of suggestions, however. First, the sug-

DISTRIBUTOR PRICES ARE NOW ACCESSIBLE!

SPECIALS!

CCA EXCLUSIVE



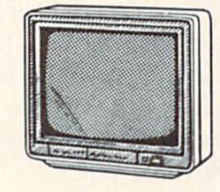
- Faster than Commodore 1541
- Half-Height Drive
- External Power Supply
- No Overheating
- 100% Commodore Compatible
- Direct Drive Mechanism
- 2 Yr. Extended Warranty
- Free Software Included

NEW FOR commodore

ENHANCER 2000 DISC DRIVE

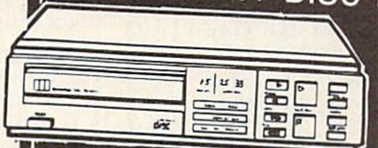
$164.95

SHARP 13" COLOR COMPOSITE MONITOR 13M31U



$144.95

LOOK! CCA NOW HAS AN EXCLUSIVE LINE OF COMPACT DISC PLAYERS!

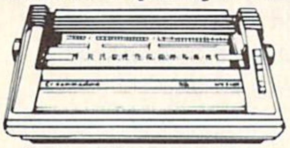


Emerson
ADC
Sherwood
Magnavox
and Many More

- Triple Beam Pick-up
- Analog Filtering
- 16 Track Programmability
- Front-Loading Disc Tray

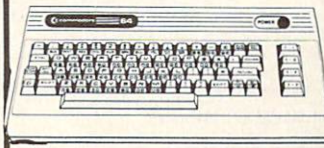
Priced From **$159.95**

commodore Letter 1101 Quality Printer



$274.95

commodore 64



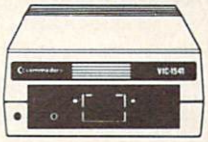
$144.95

commodore 1702 Monitor



$174.95

commodore 1541 Disc Drive



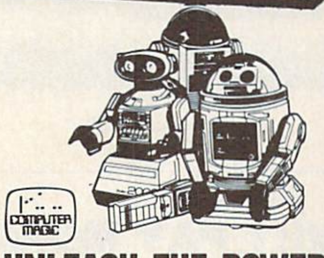
$169.95

SUMMER SOFTWARE SPECIAL! from EPYX



- Fast Load
- Breakdance
- Jet Combat Simulator
- G.I. Joe
- Summer Games

YOUR CHOICE $19.99



UNLEASH THE POWER OF YOUR COMPUTER DIRECTLY INTO THE TOMY ROBOTS.

•Includes Computer, Interface and Disk •Create, Edit, Save, Retrieve and Mix Your Programs •Robot*Link is Also a Powerful Educational Tool •For Tomy Omnibot, Omnibot 2000 and Verbot All on the Same Disk

CALL FOR PRICE

WE NOW HAVE A LARGE SELECTION OF COMPACT DISCS

CALL FOR OUR PRICES



CBS

NEW!
BILLY JOEL'S GREATEST HITS
Michael Jackson - Thriller .EK 38112
Wham! - Make It Big .....CK 39595
Barbra Streisand - Emotion CK 39480
Johnny Mathis - Right from the Heart ....CK 39601
Billy Joel - An Innocent Man ....CK 38837
Julio Iglesias - 1100 Bel Air Place ....CK 39157
Bruce Springsteen - Born in the USA ....CK 38653

Warner/Elektra Atlantic

Phil Collins - Hello I Must Be Going 80035
John Fogerty - Centerfield 25203
Eagles - Greatest Hits 1971-75 105
Al Jarreau - Breaking Away 3576
Carly Simon - The Best of 109
Prince & The Revolution - Purple Rain 25110
Madonna - Madonna.........23867
Van Halen - 1984 23985
Grover Washington, Jr. - The Best Is Yet To Come ...60215

DELOS

Joe Williams - Nothin' But the Blues 4001
Bobby Shew/Chuck Findley - Trumpets No End 4003
Mavis Rivers - It's A Good Day .4002
Maria Callas - Cherubini — Medea ....ACDOCL 201

second hearing

Eastern Jazz Ensemble - Live! ....NGS 9002
Joseph Villa - Piano - Liszt Recital...........GS 9001
Anthony Newman - Newman Plays Newman ....GS 9005

TELARC

Tchaikovsky, Cinunutti Orch. - 1812 Overture CD80041
Stravinsky, Atlanta Symph. - The Firebird ...CD80039
Beethoven, The Cleveland Orch., Symph. #3 "Eroica"...............CD80090

DENON

J.S. Bach - Trio in D Minor .C37-7093
Nancy Wilson - I'll Be a Song C38-7061
Art Farmer with The Great Jazz Trio - Ambrosia C38-7091

WE'RE PROUD OF OUR PRICES — SATISFACTION GUARANTEED

Check, Money Order, MC or VISA accepted / No additional charge for MasterCard and Visa / Call for shipping and handling information / NYS residents add applicable sales tax / Prices and availability are subject to change without notice / All factory fresh merchandise with manufacturer's warranty. Dealers Welcome. Call for recent price reductions and new rebate information. Plus shipping and handling. No returns without return authorization number.

COMPUTER Centers of America Mail Order LTD.

Reader Service No. 231

gested location of the switch on top of the computer to the right of the power LED is somewhat vulnerable to inadvertent contact during operation. Given the purpose of the switch to erase all memory, it would be disastrous for it to be pushed while the memory was full of important data. I have placed my button on the right side of the computer just above the main power switch. Second, while shopping for the switches recommended by the article, I found a smaller, less likely to be accidentally tripped switch which comes two to a package at $1.59 (Radio Shack #275-1571).

Thanks again for this excellent article and magazine! Keep up the good work.

—Joseph Bradley
Ossipee, NH

In the August issue of *Ahoy!* there were two items that raised my interest enough to write to you.

The first had to do with Dale Rupert's article on calculating pi by using the random number generator. His method is an interesting one, but there is an even simpler way. Many years ago, when I was learning FORTRAN, I wrote a program to calculate pi which is rewritten below in BASIC:

```
10 FOR I=1 TO N
20 X=RND(1):Y=RND(1)
30 IF X2 + Y2 >1 THEN 50
40 CC=CC + 1
50 NEXT I
60 PRINT 4*CC/I
```

The x and y represent coordinates of a point within a rectangular area 1 unit x 1 unit. In line 30 we check to see if the point lies within a quadrant of a circle with a radius of 1. The ratio of the area of the quadrant to the area of the square is pi/4, and, therefore, the ratio of randomly generated points within the quadrant to the points within the square should also be pi/4. Running the program above with the N=231,878 that Mr. Rupert mentions, produced pi= 3.12792448 on my machine, which does indicate a certain lack of true randomness. Perhaps it might be of interest to the readers for you to publish an article on the workings of the random number generator. I know that I would be interested.

The second item I noticed was the reply to Sean Flynn in the *S.O.S.* column concerning improving the performance of a monitor. I also use a TV/monitor (Panasonic CT-1112) and found that I could improve the clarity greatly by mixing the luminance signal with the video signal before it enters the monitor. In order to control the amount of luminance signal being mixed with the video, I first ran the luminance through a potentiometer and then, by means of a Y-connector, merged the two signals into the video input of my monitor. This has worked wonders, as the quality rivals that of the 1702 monitors I have seen. I can even read the 80-column output *PaperClip* produces! Since it is so easy and cheap to do, I would highly recommend it to anyone.

I hope this is of use.

—Theodore V. Brown
Hyde Park, NY

For Christmas I received a computer with a datasette. This computer happens to be a Commodore 16. You know the one I'm talking about, it's the one with the number 16 on it.

The Commodore 128 is also new, and merited a big feature in the August issue. What about the 16? Haven't your programmers heard of it? I think there are enough programs for the VIC and 64. What about the Commodore 16 owners of the free world?

—Larry Wedekind
Nashville, TN

It's true that we've neglected the C-16, and must regretfully report that the situation is not likely to improve. Running one C-16 program means running one less C-64, C-128, or VIC 20 program—and it's the C-64, C-128, and VIC 20 owners that make up the vast majority of our readership and potential readership. We hope you and the other "Commodore 16 owners of the free world" understand. Fortunately, there probably aren't enough of you to beat us up.

PROGRAMMING BIT BY BIT

Performing Single Bit Manipulations on Binary Numbers

BY MARK ANDREWS

Why is assembly language different from all other programming languages? There are many answers to that question. Readers of this column know by now that assembly language is the fastest-running and most memory-efficient of all programming languages. But it has another feature that we haven't examined yet. More than any other language, it gives the programmer direct and complete control over every individual bit in a computer's memory. That may not mean much to you now, but by the end of this column it will mean quite a bit. Before we begin, I'd like to throw some figures around.

There are 65,536 bytes of memory in a Commodore 64, and 131,072 in a Commodore 128. Since there are eight bits in a byte, this means that there are 524,288 *bits* in a 64, and up to 1,048,576 bits in a 128. And if you can control each of those bits individually–as easily as programmers in most other languages can handle bytes –you can wield a tremendous amount of power over your computer.

PROGRAMMING AN 8-BIT COMPUTER

Both the C-64 and the C-128 are 8-bit computers, so-called because each is built around an 8-bit microprocessor chip. The 64's main microprocessor is an 8-bit 6510 chip, and the 128's is an 8-bit chip called the 8502.

These are called 8-bit chips because the largest number they can process in one operation is 255–written 11111111 in binary notation, and therefore called an 8-bit number. To perform operations on a number larger than 255, an 8-bit computer must split the number up into 8-bit chunks, then perform the requested operation on each piece. Then, if the result of the calculation is larger than 255, it must be reassembled into a complete number.

Assembly language programmers use many tools to help them out with all this electronic cutting and pasting. And one of their most important tools is the carry bit, or carry flag, of the processor status register.

SHIFTING AND ROTATING INSTRUCTIONS

The processor status register (or P register) is one of several internal registers in the 6510/8502 microprocessor. The P register consists of eight bits, and seven of them are used as flags in various kinds of 6510/8502 operations. (The eighth bit is not used.)

The carry flag, or C flag, is one of the busiest flags in the P register. When the 6510/8502 performs arithmetic operations on numbers larger than 255, the carry flag is the flag used to keep track of carrying and borrowing operations. Also, the C flag is often used with four specialized assembly language instructions called shifting and rotating operations: ASL (arithmetic shift left), LSR (logical shift right), ROL (rotate left), and ROR (rotate right).

ASL (ARITHMETIC SHIFT LEFT) AND LSR (LOGICAL SHIFT RIGHT)

The instructions ASL, LSR, ROL, and ROR are often used in multiplication and division operations. With the help of these registers, it's easy to perform certain simple arithmetic operations, such as multiplying a number by 2. To double a binary number, you simply shift every bit to the left by one space, and fill in the vacated space with a zero. Take the binary number 0001, move every bit one space to the left, and you have 0010, or 2. Shift each bit to the left again, and you have 0100, or 4.

To divide a binary number by 2, shift bits in the other direction. Take the binary number 10000000 ($80), move each bit to the right, and you have 01000000 ($40). Shift right again, and you get 00100000 ($20).

Now let's take a closer look at how the ASL and LSR instructions work. We'll begin with ASL:

Processor Status Register

B I T S

```
 7   6   5   4   3   2   1   0

|8   7   6   5   4   3   2   1 <---- 0
v
C
```

The instruction ASL moves each bit in an 8-bit number one space to the left. If there is a remainder, it drops into the carry bit of the P register.

Following is an example of how an ASL instruction can be used to multiply a number by 2 in an assembly language program. (All the routines in this column were written using a *Commodore 64 Macro Assembler.*

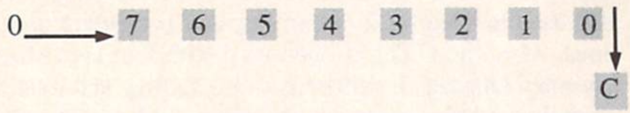
```
10 ;
20 *=$8000
30 ;
```

```
40  LDA #$40 ;REM 0100 0000
50  ASL A ;SHIFT VALUE IN ACCUMULATOR TO
 LEFT
60  STA $FB
70  .END
```

The LSR instruction is the exact opposite of the ASL instruction, as you can see from this illustration:

Processor Status Register

B I T S



As you can see, LSR can shift each bit in a number one position to the right. Bit 7, left empty by the LSR instruction, will be filled in with a zero. But Bit 0, the rightmost bit, will drop into the carry bit of the P register.

Here is an example of how the LSR instruction can be used to halve an 8-bit binary number:

```
10 ;
20 ;DIVIDING BY 2 USING LSR
30 ;
40 VALUE1=$FB
50 VALUE2=$FC
60 ;
70  *=$8000
80 ;
90  LDA #6 ;OR ANY OTHER 8-BIT NUMBER
100  STA VALUE1
110 ;
120 ;NOW WE'LL DIVIDE BY 2
130 ;
140  LDA VALUE1
150  LSR A
160  STA VALUE2
170  .END
```

ROL (ROTATE LEFT) AND ROR (ROTATE RIGHT)

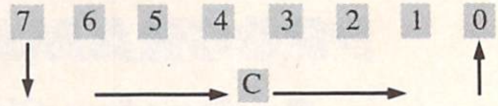
The instructions ROL (rotate left) and ROR (rotate right) use the carry bit differently from the ASL and LSR instructions.

ROL, like ASL, can shift the contents of the accumulator or a memory register one place to the left. But ROL does not place a zero in the Bit 0 position of the number being shifted. Instead, it rotates the carry bit into Bit 0, then moves every other bit one place to the left, finally dropping Bit 7 back into the carry bit. If the carry bit is set when the rotation takes place, a 1 is placed in the Bit 0 position. If the carry bit is clear, a zero goes into the Bit 0 position.

Here is an illustration of how the ROL instruction works:

The ROL ("Rotate Left") Instruction

Processor Status Register

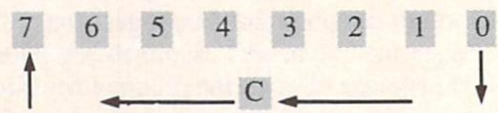


ROR works just like ROL, but in the opposite direction. It moves each bit one position to the right, rotates the carry bit into Bit 7, and moves Bit 0 into the carry bit.

Here is an illustration of the ROR instruction:

The ROR ("Rotate Right") Instruction

Processor Status Register



ROL and ROR are often used in multiplication and division routines involving numbers more than 8 bits long. For example, a 16-bit number can be doubled by combining an ASL instruction with an ROL instruction. First, an ASL instruction can be used to double the low-order byte of the number. Then an ROL instruction can be used to double the number's high byte. When the ASL instruction is used on the low byte, it will drop the leftmost bit (Bit 7) of the low byte into the C flag of the P register. Then the ROL instruction will pick up that bit and rotate it into the rightmost bit (Bit 0) of the high byte. The result of this operation will be a 16-bit multiplication!

FOUR MORE SINGLE-BIT INSTRUCTIONS

Three other instructions often used in single-bit operations are AND ("and"), ORA ("or"), and EOR ("exclusive or"). These are often referred to as *logical operators* because (logically enough) they're used in logical operations.

In 6510/8502 assembly language, the logical operator AND has the same meaning that the word "and" has in English. If one bit and another bit both have a value of one, the AND operator also yields a value of 1. But if any other condition exists—if one bit is true and the other is false, or if both bits are false—the AND operator returns a result of 0.

The results of logical operators are often illustrated with diagrams called truth tables. Here's a truth table for the AND operator:

TRUTH TABLE FOR "AND"

0	0	1	1
AND 0	AND 1	AND 0	AND 1
0	0	0	1

In 6510/8502 assembly language, the AND instruction

is often used in an operation called **bit masking**, the purpose of which is to clear or set specific bits of a number. The AND operator can be used, for example, to clear any number of bits by placing a zero in each bit that is to be cleared.

This is how that kind of bit-masking operation could work:

```
100  LDA #AA ;BINARY 1010 1010
110  AND #F0 ;BINARY 1111 0000
```

If your computer encountered this routine in a program, the following AND operation would take place:

```
    1010 1010 (CONTENTS OF ACCUMULATOR)
AND 1111 0000
-----------------------------------------
    1010 0000 (NEW VALUE IN ACCUMULATOR)
```

As you can see, this operation would clear the low nibble of $AA to $00 (with a result of $A0). The same technique would work with any other 8-bit number. Whatever the number being passed through the mask 1111 0000, its lower nibble would always be cleared to $00—and its upper nibble would always emerge from the AND operation unchanged.

THE "ORA" OPERATOR

When the instruction ORA ("or") is used to compare a pair of bits, the result of the comparison is 1 (true) if the value of either bit is 1. This is the truth table for ORA:

TRUTH TABLE FOR "ORA"

0	0	1	1
ORA 0	ORA 1	ORA 0	ORA 1
0	1	1	1

ORA is also used in bit-masking operations. Here is an example of a masking routine using ORA:

```
LDA VALUE
ORA #$0F
STA DEST
```

Suppose that the number in VALUE were $22 (binary 0010 0010). This is the masking operation that would then take place:

```
    0010 0010 (IN ACCUMULATOR)
ORA 0000 1111 (#$0F)
-----------------------------------------
    0010 1111 (NEW VALUE IN ACCUMULATOR)
```

THE "EOR" OPERATOR

The instruction EOR ("exclusive or") will return a true value (1) if one—and *only* one—of the bits in the pair being tested is a 1.

This is the truth table for the EOR operator:

TRUTH TABLE FOR "EOR"

0	0	1	1
EOR 0	EOR 1	EOR 0	EOR 1
0	1	1	0

The EOR instruction is often used for comparing bytes to determine if they are identical, since if any bit in two bytes is different, the result of a comparison will be non-zero. Here is an illustration:

```
    EXAMPLE 1                     EXAMPLE 2

    1011 0110                     1011 0110
EOR 1011 0110       BUT:      EOR 1011 0111
-------------                 -------------
    0000 0000                     0000 0000
```

In Example 1, the bytes being compared are identical, so the result of the comparison is zero. In Example 2, one bit is different, so the result of the comparison is non-zero.

The EOR operator is also used to *complement* values. If an 8-bit value is EOR'd with $FF, every bit in it that's a 1 will be complemented to a 0, and every bit that's a 0 will be complemented to a 1—like this:

```
    1110 0101 (IN ACCUMULATOR)
EOR 1111 1111
-----------------------------------------
    0001 1010 (NEW VALUE IN ACCUMULATOR)
```

Still another useful characteristic of the EOR instruction is that when it is performed twice on a number using the same operand, the number will first be changed to another number, then restored to its original value. For example:

```
    1110 0101 (IN ACCUMULATOR)
EOR 0101 0011
-------------------------------------
    1011 0110 (NEW VALUE IN ACCUMULATOR)
EOR 0101 0011 (SAME OPERAND AS ABOVE)
-------------------------------------
    1110 0101 (ORIGINAL VALUE RESTORED)
```

This capability of the EOR instruction is often used in high-resolution graphics to put one image over another without destroying the one underneath. (Yes, that's how it's done!)

THE NAME GAME

Now we're ready to take a look at a program I promised you last month: *The Name Game.* You can find it in the program listings section of this issue. It was created using a *Merlin 64* assembler, but with modifications of a few directives, it can also be typed and run on a *Commodore 64* assembler. If you own a *Commodore 64* assembler, change the ORG $8000 directive in Line 4 to "*=$8000," and change the EQU directives in Lines 6 through 12 to "=" signs. For the "TXTBUF DS 40" statement in Line 16, substitute the statements "TXTBUF =*" AND "*=*+40". Use the .BYTE directive instead of the ASC and HEX directives in Lines 18 through 31. And, instead of using asterisks to identify comments, use semicolons. The program should then work fine.

I'll wait while you type *The Name Game*, assemble it, and run it. Then come back, and I'll explain how it works.

Finished? As you've seen, *The Name Game* makes use of two operating system (or "kernal") subroutines that are built into the Commodore 64 and 128. One of these, CHROUT, is often used in C-64 programs to print characters on a computer screen. The other routine, called CHRIN, can *read* characters that are typed in on a computer keyboard. The call address of CHROUT is $FFD2, and the call address of CHRIN is $FFCF. In the *Name Game* program, the CHROUT routine is used in a loop that extends from Line 42 to Line 55, and the CHRIN routine is used in a loop that extends from Line 90 to Line 97.

Before the *Name Game* program begins, some space is set aside for a text buffer (in Line 16), and the lines of the text that will be used in *The Name Game* are listed as strings of data in Lines 18 through 31. Next there are two subroutines that will be used later on in the program. One, labeled FILL, will clear the text buffer whenever it's called. The other subroutine, called PRINT, uses the Commodore 64 CHROUT routine to print messages on the screen.

As you type, assemble, and run *The Name Game*, you may notice that it uses its text buffer for lines that are typed in at the keyboard, not for lines that are called from RAM. Some kind of buffer is obviously needed for typed-in lines, since the computer must hold them in its memory long enough to do some comparison and printing operations. Another text buffer could have been set up for the lines stored in RAM, but it would have accomplished no real purpose except consume more memory and take up more processing time.

RUNNING THE PROGRAM

The main section of the *Name Game* program starts at Line 57 with a routine that prints the program's title on the screen. The next two routines print the line "Hello, what is your name?"

After this question is asked, the program clears the text buffer and waits for the player to type in a response. As he does, each character is placed in the text buffer. That's all that happens until the player stops typing characters and hits a carriage return.

Once a carriage return is typed, the program examines the characters that have been stored in the text buffer to see whether they spell the name GEORGE. If the player has not typed in the name GEORGE, the computer prints "GO AWAY, [NAME], BRING ME GEORGE!" Then the game starts again. Computers are very patient, so this process will continue until the player weakens and types the name GEORGE. Then the computer will print "HI, GEORGE," and the game will end.

That's all there is to *The Name Game.* 'Bye, George! □

SEE PROGRAM LISTING ON PAGE 137

THE 1541 DISK DRIVE CLEANING UTILITY

By John DeRosa

When was the last time I cleaned my disk drive? How many times have I used my cleaning disk? Is it time to throw out that disk and buy a new one? How do I spin the cleaning disk without causing an error that results in chattering my drive? This utility program will answer these questions, as well as help you keep your 1541 drive head clean. The program is for use with any cleaning kit that requires spinning a special cleaning disk.

The program begins by inputting the date (variable DT$), the amount of time the disk is to spin while cleaning (SC), the number of times the cleaning disk was used (NC), and the maximum time the disk can be used (NM). This information comes with the cleaning disk instructions. The input can be from either a data file stored on the program disk, or the program will assume values. You can change the input data and save it on a disk prior to cleaning. If the number of times the cleaning disk was used exceeds the maximum, a warning is displayed and you are asked to buy a new cleaning disk.

After the input is read, the drive head will be moved to a different track from the last cleaning to assure that the same part of the cleaning disk is not used over and over. You are then prompted to insert the cleaning disk. The disk will spin for the desired amount of time, and a tone is played while it is spinning. The drive motor is turned on by setting the third bit of location $1C00 (hexadecimal) in the disk drive memory using the Memory-Write command. (See lines 1800 to 1840 of the program listing.) The motor is turned off by resetting this bit. Since the disk operating system is not accessed to turn the drive on and off, the drive will not chatter. When cleaning is complete the drive stops spinning and the program ends by asking you to insert a data disk if you want to update the input information (new date, increment number of times cleaning disk was used, etc.) and store it on the disk so it can be accessed the next time you clean your drive.

To answer the question of how often you should clean your drive, some cleaning kit manufacturers recommend once a week with heavy use. I think this is excessive, and it may even be harmful to expose the drive head to the abrasive cleaning disk this often. I average two hours a day on my C-64 and have been cleaning my 1541 disk drive once every two months without any problems. □

SEE PROGRAM LISTING ON PAGE 140

CADET'S COLUMN

For Beginning Users of the Commodore Computers

BY CHERYL PETERSON

Last month's column covered disk drives, and I'm going to continue the subject for a while. Those of you who didn't have one went out and bought one, didn't you?

If you did, you may have run into the 1541's spurious error messages, such as "File not found" (when you know it's there!) or other general read or write errors. The reason they're spurious messages is that they only occur after the drive has been on for about fifteen minutes.

The disk seems to be fine if you use it when you first power up the system. Unfortunately, by the time the drive starts messing up, you've got 15 or 20 minutes' worth of work done and are trying to perform a SAVE.

Most folks figure there's something wrong with the disk and run to the phone to call the manufacturer and complain. The manufacturer offers to send another disk, and when it arrives it has the same problem! We're talking maximum frustration here. Especially for new owners, who frequently assume they did something wrong.

Not knowing what to try next, you usually give up and take the thing off to the repair shop, where it's impossible to duplicate the problem. Why? Once you power down and wait 15 minutes, everything appears to be fine again. The service technician gives it back, along with a bill for who knows how much. When you get it home, the same thing happens again. Sound familiar? For most people using 1541's, it already does or soon will.

KEEPING THINGS COOL

The problem is caused by overheating. Some of the chips in the 1541 are particularly heat sensitive and drop dead when they get hot. Since the damage isn't permanent, once they cool down the chips work fine once again.

There are two ways to deal with the problem: take the drive apart and replace the chips with heat resistant ones (very expensive) or buy a fan to keep the drive cool. I have a little experience with two of the several different companies that sell cooling fans. (See accompanying list.)

Uni-Kool makes a nice little fan that sits on top of the 1541 and blows the cool air directly into the drive. Though it's a little more expensive than some of the others, it has one serious advantage: a replaceable filter. If you use your drive in a room with lots of cigarette smoke or dust particles, a fan blows all those contaminants right on your disks. As far as I know, the Surgeon General hasn't required cigarette pack warnings that say "Smoking may be hazardous to computer disks!", but contaminants can mess up both drive and disks. So the changeable filters in the Uni-Kool are a wise precaution.

Innovative Organizers, Inc. markets plastic stands that cover the disk drive and will support a printer on top. A companion $29.95 fan can be mounted inside the cover to cool the drive. I've been using the setup for a few weeks now with a drive that was having problems, and it's behaved beautifully. Being a stickler for clean drives, I rustled up an old pair of silk stockings and tied one around the air intakes for filtering purposes. (A pair of nylon panty hose would work just as well and be considerably cheaper.)

In fact, as a temporary measure, I borrowed a $5 Archer fan that a friend had bought at Radio Shack and set that on the stocking. Except for the radio interference caused by the fan's coil, it worked fine. The interference caused the monitor's picture to waver so badly that within two minutes of sitting down in front of it, even seasoned sailboaters would be seasick. To show you how conventional we are around here, the wavering problem was solved by putting an old beat up saucepan over the fan. This left enough clearance for the air to get through and still dampened the radio signals, though I admit it looked awfully strange. An empty coffee can also worked.

What I'm trying to point out is that you don't have to spend more money than you have. These three different ways to solve the same dilemma cost from $7 for the homemade solution to $41.95 for the Uni-Kool fan to $50 for the printer stand/fan combination.

QUICK FIX

Every now and then a program just won't load even though you're using a fan. Nothing you try seems to get things going again. Although your drive may have gone out of alignment, this isn't likely unless it's been fairly erratic for a while. Usually, drive alignment problems start with the drive occasionally having problems loading and things just seem to keep getting worse.

If this isn't the case and the drive just suddenly won't load at all, try taking the disk out and leaving the door open while you load thin air (thick air if you live at sea level like I do). Type your usual LOAD "*",8 and let the

drive bang around until it puts up the "FILE NOT FOUND" error message. Turn the whole system off and then back on and try to load your program. Chances are good that it will work.

This problem is caused by the head being left in an unusual position. The 1541 head usually stops itself somewhere near the directory tracks in the middle of the disk after it does a read or write. Sometimes it doesn't get set properly before you turn off the machine and the next time you power up and look for a program the head gets lost. Banging it around with a fake LOAD command will usually reset it.

WHERE TO PUT IT ALL!

I have a few more organization tips for you this month. If you've taken any of my earlier suggestions, these new ones will probably come in handy. You'll need somewhere to put the magazines, books, software, and assorted peripherals that you'll be collecting as you make full use of your computer. I recommend metal shelves that you put together yourself for $15 to $20. Sears, K-Mart, and many local department stores run specials on them about every two months. Because you put them together yourself, the shelves can be arranged at intervals to suit you. Being heavy duty, they can support quite a bit of weight. (Paper gets heavy if you collect enough of it.)

GETTING A DESK

One thing this will do is clear your desk of unneeded junk. What do you mean you don't have a desk? If your computer is still occupying the dining room table, you should move it. For two reasons! Disrupting the family harmony by usurping the main meeting place will not endear you to your potential computer widow/er. And there are too many things that can get spilled on a C-64 that will kill it.

In most homes, the husband and kids use the computer and the poor mother is left bereft of a table to serve dinner on. Even in our family, where I was the one using the computer, we went through the "get that #$%!! computer off my dining table" fight, but with a different twist. Hubbie didn't want to buy a desk; he wanted to make one. But getting him to go out and buy the materials was like trying to drain the ocean with a straw. Eventually it got done, though.

For those who want a spacious desk with plenty of drawer space, try buying a door (yes, your standard 8x3' particle board covered with plywood door will do) and a couple of 28-inch deep, 2-drawer filing cabinets. If you get them at a closeout or bankruptcy auction, they'll probably run you about $50. For another $20 to $30, you can get a can of wood finisher that will make your cheap plywood look like something more expensive.

Sears typically sells "computer" desks for $90 plus $35 for an optional monitor stand. The desk measures 5x3' and will barely accommodate a computer, disk drive, operating manuals, and a printer. By the time you add a diskette file and a few choice software packages, you don't have enough room left for the spare telephone you'll be needing. (Besides coming in handy when you need software support from your local user group friend, the modem you'll want to use hitches into the phone. And when you are seriously involved in a project, you won't want to leave your desk every time the phone rings.)

You'll also want extra room for the midnight snacks and coffee needed to keep going on those long nights when you're debugging programs from your favorite magazines. You might be able to survive without the coffee maker on the desk, as it gives you a good reason to goof off for a few minutes.

I must admit, in addition to our door desk, I did buy a "computer" desk. (My husband stole the door for his computer.) As you can see from the photo (page 110), it was worth the $200 I paid. The shelves hold plenty of software and diskette files and support a couple of monitors and two computers. Even so, I have my printers on a separate stand to the right. It came unassembled in a box labeled "TV stand for use with VCR, can also be used for microwave ovens or computer printers." It cost me $40 at a SuperX drug store and holds three printers, paper, tractor feed labels, extra ribbons, and manuals. Businesses are paying office supply stores upwards of $150 for printer stands that hold one printer and some paper.

Unless you intend to get very serious about using your Commodore, or you really need a desk anyway, or you have a lot of money to blow, I wouldn't recommend spending more than $100 for a desk. You can get a card table with S&H Green stamps for about 15 books. This will at least allow the family to share dinners in the dining room again.

If you have your heart set on a really neat "computer" desk, I recommend checking with your local furniture

FANS FOR 1541

COM COOL FAN
Big Bytes
1301 Boardman-Poland Rd.
Poland, OH 44514
In OH 216-758-0009; rest of USA 800-638-2617
$49; with surge protector $69 (plus $2.50 shipping)

COMPU-COOL
P.O. Box 12094
San Antonio, TX 78212
$39.95 (plus $2 shipping)
(similar to Uni-Kool)

UNI-KOOL
909 Williamson Loop
Grants Pass, OR 97526
(503) 476-1660
$39.95 (plus $2 shipping)

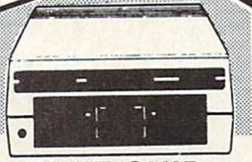
MUFFIN FAN
Bytes & Pieces, Inc.
550 N. 68th St.
Wauwatosa, WI 53213
(414) 257-1214
$54.95; with surge protector $79.95 (plus $2 shipping)

P-D-1, $19.48
CF-1 COMPUTER FAN,
$29.95 (shipping additional)
(Innovative Organizers, Inc.)
P.O. Box 108
Markleville, IN 46056
(800) 233-3645 ext. 15

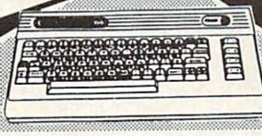
WORLD TRADE

COMPUTER WHOLESALE OUTLET

Call 1-800-722-4002



BLUE CHIP DISK DRIVE
$159

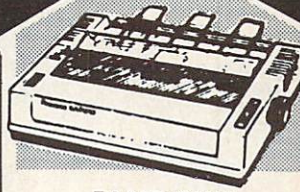


COMMODORE C-64 COMPUTER
$139

COMMODORE 128 Soon To Be Available Call

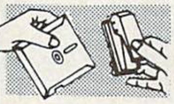


TEKNIKA COMPOSITE MONITOR
$159



BLUE CHIP PRINTER With XETEC INTERFACE
$259

- Fast 130 char./sec. near letter quality
- Italic super script
- Tractor & friction.



NEW! NIBBLE NOTCH

Turns your 5½" Single Disk into A 2 Sided Disk **7.95**

DISK DRIVES

COMMODORE 1541	$165.00
ENHANCER 2000	$159.00

EPYX SOFTWARE

Ballblazer	25.95
Barbie	19.95
Breakdance	25.95
Fast Load	CALL
G.I. Joe	19.95
Summer Games	25.95
Summer Games II	25.95
9-5 typing	25.95
Rescue on Fractalus	25.95

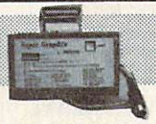
GAMESTAR SOFTWARE

Baseball	19.95
Football	19.95

ACCESS SOFTWARE

Beach head	24.95
Beach head II	29.95
Mach 5 w/util software	24.95

INTERFACES



SUPER GRAPHIX 8 K buffer, 10 printing modes **69.95**

Super Graphix Jr.	49.95
Tymac Connection	79.95
G-Wiz	49.95
PPI	49.95

MODEMS



Video 7	59.95
Mitey Mo (1 hour free compuserve time	$ 63.00
West Ridge	79.95

MONITORS

Commodore 1702	169.95
Sharp 13" Color Compsite	$144.00
Sharp green 12"	59.95
Sharp amber 12"	69.95
Teknika RGB/ Composite 13"	$269

DISKETTES



Nashua (box of 10)	10.95
Elephant (box of 10)	11.95
Scotch (3M) (box of 10)	12.95
Scotch Head Cleaning Kit	9.95

PRINTERS

Blue Chip M120/near letter quality Tractor/friction 120cps/60cps	189.00
Blue chip M150/10 Tractor/friction 130cps	199.95
Blue chip M150/15	269.95
Star sg10 tractor/friction 130cps	205.95
Star sg15 tractor/friction 130cps	359.95
Juki 6000 letter quality	219.95
Juki 6100 letter quality	429.95
Okimate 10 color printer	219.95
Legende 808	159.95
Legende 880	219.95
Legende 1080	239.95
Legende 1380	279.95
Legende 1385	349.95
Cardco LQ3 letter quality	189.95

PAPER & RIBBON



Delivery on paper included in price

paper size 9½" x 11"

1000 x 3 microperf	60.00
2700 microperf	55.00
Ribbon bluechip m120/m150	9.95
Ribbon SG10/SG15/ okidata 82a	3.95
Ribbon olivetti et/tes	12.95
Ribbon silver reed	12.95
Ribbon 3m 400/600/800	12.95
Ribbon commodore 1101 (6 per pak)	19.95
Ribbon okimate multi color	7.95

JOYSTICKS



Wico boss	10.50
Wico bat	$ 14.95
Wico threeway	19.95
Gim 1000-left/right firing button	9.95

DISK FILES



10 disk holder	6.00
60 disk holder (stackable)	12.95



SURGE PROTECTORS
19.99 & Up

WORD PROCESSING

Fleet system 2 w/70,000 word spellpak	54.95
Font master	14.95
Creative writer	34.95

TIMEWORKS

Data manager 2	39.95
Evelyn Wood dynamic reader	19.95
Work writer w/spell check	37.95
Swiftcalc w/sideways	37.95
Cave of the Word Wizards	27.95

ATARI SOFTWARE ALSO AVAILABLE

CBS SOFTWARE



Grand prize $10,000 cash for tuition. 1st prize $1,000 educational software 2nd prize $100 each for school supplies Plus...with proof of purchase you will get $10 rebate on CBS software's Easy Key programs.

Many ways to say I Love You	18.95
Letter go-round	
Astro-grover	18.95
Big bird's funhouse	18.95
Ernie's magic shapes	18.95
Big bird's special delivery	18.95
Sea horse hide'n seek	9.95
Dream house	24.95
The sea voyagers	24.95
Light-waves	14.95
Fleet feet	14.95
The argos expedition	23.95
Weather tamers	14.95
Forcast	32.95
Time bound	14.95
Math mileage	19.95
Workfinder	23.95
Dinosaur dig	24.95
Coast to coast	24.95
Mastering the act	54.95
Mastering the sat	54.95
Success with math series	17.95 ea.
Success with algebra series	24.95 ea.
Felony	24.95
Murder by the dozen	24.95
Adventure master	32.95
Pathwords	23.95
Muppet learning keys	79.95

WHEN IN THE NEW YORK AREA VISIT OUR 6 BIG L STORES.

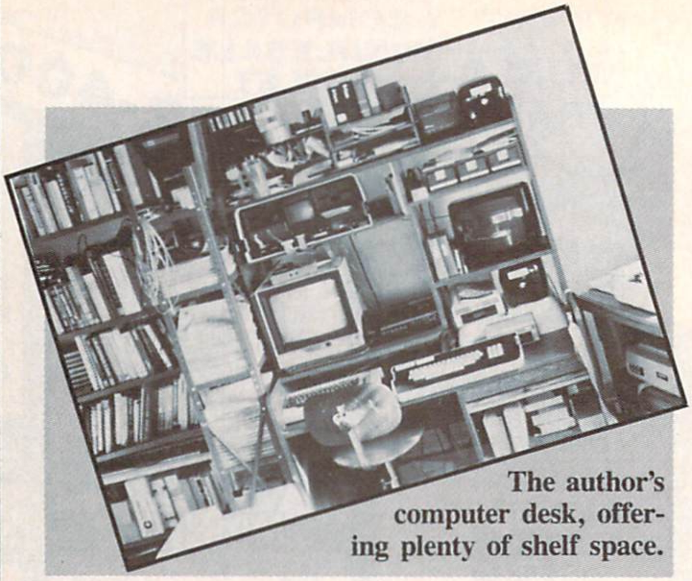
• West Babylon • New Hyde Park • Syosset • West Hempstead • Patchogue • Bayside

CALL TOLL FREE 1-800-722-4002. IN NEW YORK CALL 718-895-2388

MasterCard VISA C.O.D.

Check, Money Order, MC or VISA accepted/Call for shipping and handling information/NYS residents add applicable sales tax/Prices and availability are subject to change without notice/All factory fresh merchandise with manufacturer's warranty. Dealers welcome. Call for recent price reductions and new rebate information.

FOR INFORMATION & CUSTOMER SERVICE CALL 1-718-895-2388.



The author's computer desk, offering plenty of shelf space.

store, Service Merchandise, Best, or other bargain store not specializing in computer accessories. A local Levitz furniture store was running a sale on a computer desk very similar to mine for $80 around Father's Day, so be sure to check ads in the local paper for non-computer stores. (I won't even mention that no one in town was featuring computer equipment or accessories in their Mother's Day ads.)

And while you're at it, don't overlook auctions. Besides filing cabinets, businesses usually have desks, shelves, comfortable chairs (sometimes with wheels), and occasionally used computer monitors.

SHORT PROGRAM FOR MONOCHROME MONITORS

This short piece is for those who haven't gotten a color monitor and are making do with a black and white television or monochrome (green or amber) monitor. If run before you boot the programs you use, it will set the background and border colors to black increasing the sharpness of contrast and making it easier to read the letters.

Many commercial programs will reset the memory locations that control these parameters, so I can't guarantee it will help in every instance. When doing programming or entering BASIC programs from books or mags, it really saves the eyes. It isn't a bad idea to run it with a color monitor, since the sharper contrast is less likely to leave you with a headache.

```
5 PRINT "SCREEN TO BLACK"
10 POKE 53280,0
20 POKE 53281,0
25 PRINT "DONE"
30 END
```

Since this is such a short program, the two middle lines, 10 and 20, can be entered directly if you want to avoid having to load and run the program. I always had trouble remembering the POKE locations. This program saw a lot of use before we switched to a color monitor. ☐

COMMODARES

PROGRAMMING CHALLENGES

BY DALE RUPERT

Each month, we'll present several challenges designed to stimulate your synapses and toggle the bits in your cerebral random access memory. We invite you to send your solutions to:

Commodares, c/o *Ahoy!*
P.O. Box 723
Bethel, CT 06801

We will print and discuss the cleverest, simplest, shortest, most interesting and/or most unusual solutions. Be sure to identify the *Name* and *Number* of the problems you are solving. Also show sample runs if possible, where appropriate. Be sure to tell what makes your solutions unique or interesting, if they are.

Programs on diskette are welcome, but they must be accompanied by listings. You must enclose a stamped, self-addressed envelope if you want any of your materials returned. Solutions received by the middle of the month shown on the magazine cover are most likely to be discussed, but you may send solutions and comments any time. Your original programming problems, suggestions, and ideas are equally welcome. The best ones will become *Commodares*!

PROBLEM #22-1: MAKING CHANGE

Jim Speers (Niles, MI) suggested this challenge. Given an amount of money, say $10.00 or less, and a number of coins, find a distribution of quarters, dimes, nickels, and pennies equal in value and quantity to the specified amounts. If there is no solution, this should be stated. Five coins totaling $1.10 would be four quarters and a dime. Two coins totaling forty cents is impossible. We will present Jim's solution in a later issue.

PROBLEM #22-2: SIMPLE COMPARISON

How simple can a problem be? The user provides two sets of ten numbers each. The computer determines whether the two sets contain the same numbers or not. The numbers for each set are contained in a DATA statement and of course may be in any order. What is the fastest way to do this?

PROBLEM #22-3: MEASURED DECIMALS

Ken Karow (Chicago, IL) suggested a problem similar to this. The user enters a decimal number of inches. The computer converts this decimal to a whole number plus a lowest-terms fraction, rounded to the nearest sixty-fourth of an inch. For example, if the user enters 8.375, the computer responds "8 and 3/8." If the user enters 23.016, the computer responds "23 and 1/64."

PROBLEM #22-4: TERNARY TURNABOUT

The user enters a number in base ten, and the computer displays its ternary (base three) equivalent. The ternary system uses numerals 0, 1, and 2. For example, 100 in base ten is 10201 in base three.

This month we will look at readers' solutions to the June *Commodares*, from balancing billiard balls to a one line skiing game. Many variations on the theme of *Problem #18-1: Cycling Function* are listed below.

```
1 REM
2 REM VARIOUS READERS' SOLUTIONS TO
3 REM PROBLEM #18-1 : CYCLING FUNCTION
4 REM
10 DEF FNC(D)=D*SGN((99-X)/(X-1))
11 DEF FNC(D)=INT(X/100)*-105+5
12 DEF FNC(D)=D+10*(X=100)-10*(X=0)
13 DEF FNC(D)=D+10*(X>95)-10*(X<5)
14 DEF FNC(D)=D+(2*D*(X=0 OR X=100))
15 DEF FNC(D)=5+10*(X>95 OR D<5)-10*(X<5
)
16 DEF FNC(D)=D+(2*D*((X<=0)OR(X>=100)))
17 DEF FNC(D)=-5*(D=5)+5*(D=-5)+10*(X=10
0)-10*(X=0)
18 DEF FNC(D)=D*((((X=100)+(X=0))*2)+1)
20 D=5:X=0
30 S=X/5:PRINTSPC(ABS(S))X:X=X+D:D=FNC(D
):GOTO30
```

The programmers responsible for lines 10 through 18 are Steve Kendrick (Bakersfield, CA), Don Maloney (Manchester, CT), Tim Hedrick (Canoga Park, CA), Tim Williams (Henderson, KY), Dennis McGrath (Chicago, IL), David Hoffner (Brooklyn, NY), Sheldon Wotring (Palmerton, PA), Scott Macnab (Woodburn, OR), and James Borden (Carlisle, PA) respectively. You may compare these with the general solutions from Jim Speers (Niles, MI) in the July edition of *Commodares*. Jim originally submitted the problem.

Line 30 is modified from the original problem at the suggestion of Steve Kendrick so that the cyclic nature of the numbers is more apparent. One of these solutions cycles from 0 to 100 then starts again at 0. Can you find it before you run the program? Line 18 defines FNC the first time the program is run. After you have run the program once, you may delete line 18. Run it again and line 17 will be the definition of FNC. Repeat this process until line 10 is the only DEF statement.

Tim Hedrick attempted to solve this problem using the cosine function which itself is cyclic. He wrote:

```
10 DEF FNC(D)=D+10*INT(COS(X*PI/199))
```

where PI has the value 3.14159265. This function stepped from 0 to 100 and back to 0. However, it then continued into negative numbers instead of reversing again as it should. For X=0, INT(COS(X*PI/199)) should return a value of 1, but instead it gives 0. Tim suggested a round-off error as the culprit. That's exactly what it is.

If you add 1E−9 to the cosine function before using the INT function, everything works as expected. Try this to see for yourself:

```
10 DEF FNC(D)=D+10*INT(COS(X*PI/199)+1E-
9)
```

All readers who sent solutions to *Problem #18-2: Billiard Balls* deserve a word of praise. Most of the solutions were very lengthy, but it was a very difficult problem. The two listed below from Russell Wilson (Fayette, AL) and Charles R. Goldberg (Lambertville, NJ) are significantly shorter than the others because of the additional "pre-computing" analysis performed by the writers of these programs.

```
1 REM
2 REM SOLUTION TO PROBLEM #18-2 :
3 REM                 BILLIARD BALLS
4 REM BY RUSSELL WILSON
5 REM
110 DIMC$(12,2),S$(3):X$="":W$(1)="HEAVY
 ":W$(2)="LIGHT "
120 FORI=1TO12:FORJ=1TO2:READC$(I,J):NEX
TJ,I:FORI=1TO3:READS$(I):NEXTI
130 DATA BBR,BBL,LBL,RBR,LRB,RLB,LLB,RRB
,RBL,LBR,LLL,RRR,RLR,LRL,RRL,LLR
135 DATA BRR,BLL,BLR,BRL,RBB,LBB,BRB,BLB
140 DATA 2 3 4 6  + 5 7 8 11,4 6 7 10 +
3 8 9 12,2 5 6 8  + 1 7 9 10
150 PRINTCHR$(147);"REMEMBER BALL# (1-12
) AND IF IT IS HEAVY OR LIGHT.":PRINT
160 PRINT"ENTER L=LEFT HEAVY":PRINTTAB(6
);"B=BALANCED"
165 PRINTTAB(6);"R=RIGHT HEAVY":PRINT
170 FORI=1TO3:PRINTS$(I):PRINTTAB(7);:IN
PUTR$:X$=X$+R$:PRINT:NEXTI
180 FORI=1TO12:FORJ=1TO2:IFX$=C$(I,J)THE
N200
190 NEXTJ,I
200 PRINTCHR$(18);I;CHR$(157);" BALL IS
";W$(J)
```

```
1 REM
2 REM SOLUTION TO PROBLEM #18-2 :
3 REM                 BILLIARD BALLS
4 REM BY CHARLES GOLDBERG
5 REM
10 D$="HEAVYLIGHTWHAT?":PRINT"[CLEAR][DO
WN] ENTER DEFLECTION"
20 PRINT"[DOWN][DOWN] LEFT----N----RIGHT
"SPC(30)"O"SPC(39)"N"SPC(39)"E"
30 INPUT"[DOWN]1-2-4-5  [s Q] 7-8-10-11"
;A$(0)
40 INPUT"4-5-7-11 [s Q] 2-3- 6-12";A$(1)
50 INPUT"6-8-9-11 [s Q] 5-7-10-12";A$(2)
60 FORU=0TO2:B=B+SGN(ASC(LEFT$(A$(U),1))
-78)*3^U:A$(U)="N":NEXT:C=ABS(B)
70 E=.5+(C=10):FORU=0TO4:E=E+(C=INT(2*1.
585^U)):NEXT:D=INT(C^2-ABS(C)*13)
80 PRINT"[DOWN]BALL";C;"IS ";MID$(D$,(3.
5+2.5*SGN(B*SGN(E)))*(D<>0)^2-11*(D=0),5
)
90 B=0:GOTO 30
```

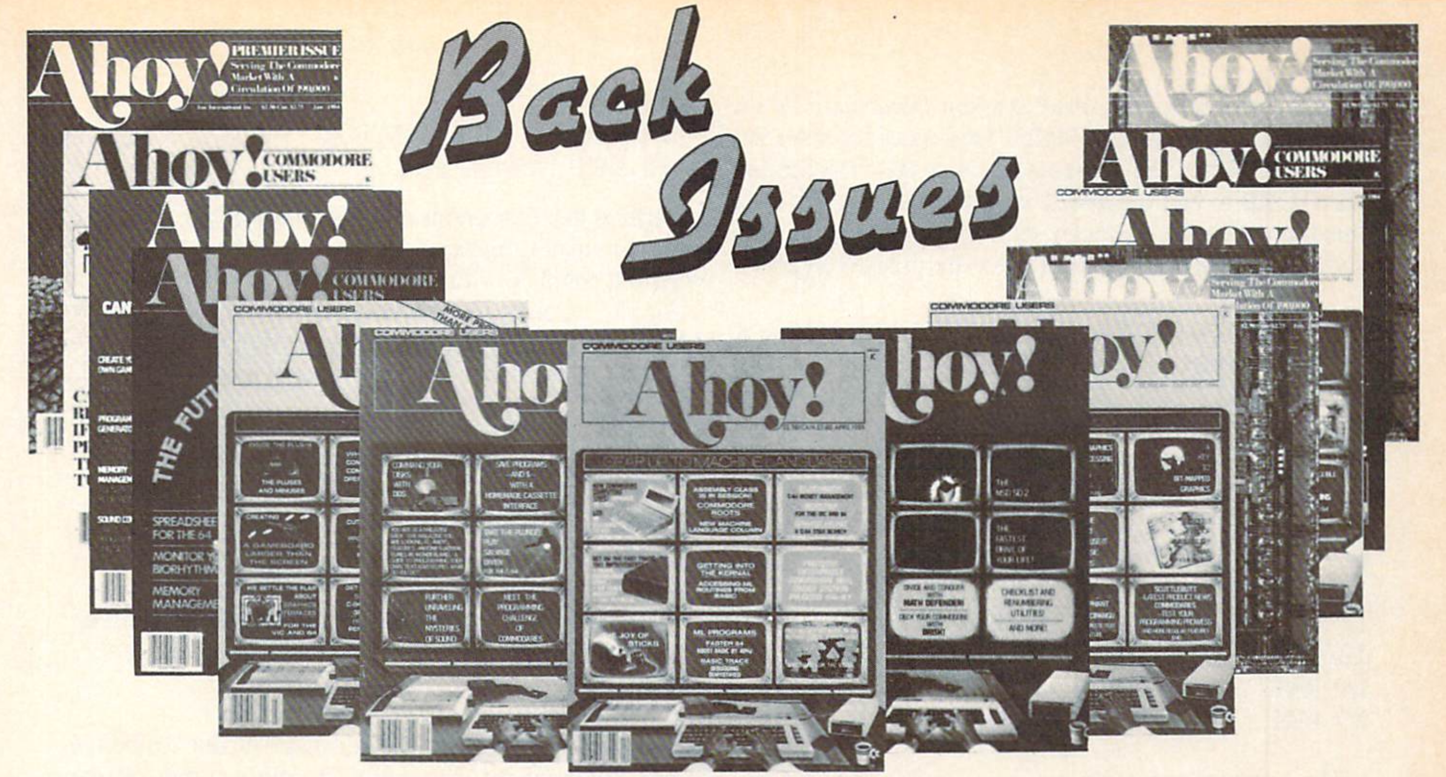
Mr. Goldberg included a discussion of the ternary (base−3) numbering system on which his solution is based. I recommend Martin Gardner's *Sixth Book of Mathematical Games from Scientific American* (Scribners, 1971) for an interesting discussion of the ternary system and this problem.

Besides Wallace Leeker (Lemay, MO) who submitted the problem and a solution, other billiard ball solutions came from Jacqueline G. Callaway (Orange Beach, AL), Steve Feld (New York, NY), Dennis McGrath (Chicago, IL), David Hoffner (Brooklyn, NY), James Borden (Carlisle, PA—a COMAL solution), Donald Maloney (Manchester, CT), and Tim Hedrick (Canoga Park, CA).

Only three people sent solutions to *Problem #18-3: Scroll Ski.* These are one or two line downhill skiing games using the scrolling feature of the screen to provide some of the animation. A brief description of each should suffice. You must type most of these programs carefully and use all possible abbreviations.

The program from Sheldon Wotring (Palmerton, PA) uses the "<" and ">" keys to move the skier. The final GOTO without an argument goes back to line 0.

```
0 PRINTTAB(T)"H":X=PEEK(197):T=T-(X=44)+
(X=47)-(T<1)+(T>38):PRINTTAB(RND(1)*39)"
[DOWN][DOWN][DOWN][DOWN][DOWN]^[UP][UP][
UP][UP][UP][UP]":GOTO
11 REM
12 REM SOLUTION TO PROBLEM #18-3 :
13 REM                 SKI SCROLL
14 REM BY SHELDON C. WOTRING
```



ISSUE #1–JAN. '84 $4.00 64 v. the Peanut! Telecommunications! Tales of the Commodore! And ready to enter: Multi Draw 64 Graphics System! Interrupt Music Maker/Editor! A Peek at Memory! Sequential Files!	**Sorry...Issues #1, #2, and #7 sold out! Reprint editions (programs and feature articles only) available for $4.00 each.**	**ISSUE #12–DEC. '84 $4.00** Buyer's guide to printers! 1525 printer tutorial! Custom characters! User Guide to KMMM Pascal! Diving into BASIC! And ready to enter: Construction Co.! Space Patrol! Cross Ref!	**Use the form below to order while limited supplies last!**
ISSUE #2–FEB. '84 $4.00 Illustrated tour of the 1541! Artificial intelligence! Synapse's Ihor Wolosenko interviewed! And ready to enter: Music Maker Part II! Screen Manipulation! Night Attack! Relative Files!	**ISSUE #7–JULY '84 $4.00** MSD dual disk drive! Database buyer's guide! Training your cursor! Screen displays! File Sleuth! Users Groups! And ready to enter: Renumbering! Checklist! Math Defender! Brisk!	**ISSUE #13–JAN. '85 $4.00** VIC/64 OS exposed! Sprites! 1541 device # disconnect switch! Ghostbusters! And ready to enter: Ultra Mail! Music Tutor! Alice in Adventureland! Midprint! To the Top! Tape/Disk Transfer!	
ISSUE #3–MAR. '84 $4.00 Anatomy of the 64! Printer interfacing for VIC & 64! Educational software series begins! And ready to enter: Address Book! Space Lanes! Random Files on the 64! Dynamic Power!	**ISSUE #8–AUG. '84 $4.00** Choosing a word processor! Computational wizardry! Creating your own word games! Sound on the 64! And ready to enter: Micro-Minder! Directory Assistance! The Terrible Twins!	**ISSUE #14–FEB. '85 $4.00** Printer interfacing! Multicolor sprites! Modems! Bulletin boards! Theory of game design! And ready to enter: Futurewar! Fontasia! VIC Eraser! Insurance Agent! Flankspeed! Telelink 64!	**ISSUE #18–JUNE '85 $4.00** Music & graphics entry systems! How modems work! Inside the 6510! And ready to enter: Quad-print! Mapping 4.4! Towers of Hanoi! Speedy! Duck Shoot! The 6510 Simulator!
ISSUE #4–APR. '84 $4.00 Petspeed and Easy Script tutorials! Printer interfacing continues! Laserdisc! And ready to enter: Apple Pie! Lunar Lander! Name that Star! Lower case descenders on the 1525 printer!	**ISSUE #9–SEPT. '84 $4.00** Program your own text adventure! Build a C-64 cassette interface! Video RAM! And ready to enter: Salvage Diver! DOS! Sound Explorer! The Castle of Darkness! Base Conversions!	**ISSUE #15–MAR. '85 $4.00** Making multiscreen gameboards! Inside the Plus/4! Commodore DOS! And ready to enter: Old Routine! Programmable Functions! Automatic Line Nos.! Home Budget! Salmon Run!	**ISSUE #19–JULY '85 $4.00** PROM programming! 3-part harmonies on VIC/64! Speeding pixels! And ready to enter: Auto-Append! Script Analysis! Wizard of Im! Lucky Lottery! Brainframe! Etch! Printat!
ISSUE #5–MAY '84 $4.00 Future of Commodore! Inside BASIC storage! Memory management on the VIC & 64! Guide to spreadsheets! And ready to enter: Math Master! Air Assault! Biorhythms! VIC Calculator!	**ISSUE #10–OCT. '84 $4.00** C-64 graphics programs! Bit-mapped graphics! Joystick programming! And ready to enter: VIC 40 Column Operating System! BAM Read & Print! Emerald Elephant! Lawn Job!	**ISSUE #16–APR. '85 $4.00** Assembly language column begins! Programming the joystick! 1541 disk drive alternatives! And ready to enter: Hop Around! Faster 64! Booter! Elecheck! BASIC Trace! Space Hunt!	**ISSUE #20–AUG. '85 $4.00** Inside the 128! Read-world simulations! Sound effects! And ready to enter: Windows! Formatter! Sound-a-Rama! Screen Dump! Selectachrome! Disintegrator! Fidgits! Gators N Snakes!
ISSUE #6–JUNE '84 $4.00 Game programming column begins! Program generators! Rupert on inputting! Memory management continues! And ready to enter: Post Time for the 64 & VIC! Alpiner! Sound Concept!	**ISSUE #11–NOV. '84 $4.00** Music programs & keyboards for the 64! Graphics feature continues! And ready to enter: PTE word processor! Block Editor! Alternate Character Set for the 64! The Tunnel of Tomachon!	**ISSUE #17–MAY '85 $4.00** Disk drive enhancements! Install a reset switch! Assembler escapades! And ready to enter: Super Duper! Two-Column Directory! DSKDU! Raid! DOS Plus! Font Editor! Tile Time!	**ISSUE #21–SEP. '85 $4.00** Inside the 1571 drive and 128 keyboard! Sprite programming! And ready to enter: Fastnew! Go-Lister! File Lock! Dragon Type! Superhero! Auto-Gen! Moxey's Porch! Fish Math!

Use coupon or facsimile. If ordering more than three issues, list choices on separate sheet.

If you have a modem and want more complete information on any of the back issues listed, call *Ahoy!*'s Bulletin Board Service at 718-383-8909.

Ahoy!
BACK ISSUES
ION INTERNATIONAL INC.
45 West 34th Street
Suite 407
New York, NY 10001

Please Send Me The Following:

_____ Copies of issue number _____
_____ Copies of issue number _____
_____ Copies of issue number _____

Enclosed Please Find My Check or Money Order for $__________

(Outside the USA please add $1.00 for every copy)

NAME ____________________
ADDRESS ____________________
CITY __________ STATE __________ ZIP CODE __________

In the program from Paul Dawson (Waitsfield, VT), the COMMODORE and SHIFT keys move the skier left and right. When the game ends, press RUN/STOP and RESTORE to end the WAIT in line 0. The CHR$(8) in line 0 prevents the character set from changing when the COMMODORE and SHIFT keys are pressed together.

```
0 P=PEEK(653)+1:Z=Z+1:PRINT"[HOME][BLACK
]SCORE"Z:POKE214,8:PRINTCHR$(8):PRINTTAB
(A)"[WHITE]"A$:IFPEEK(1424+A)=30THENWAIT
0,0
1 POKE214,23:PRINT:PRINT"[GREEN]^"TAB(RN
D(0)*35)"^^^^"TAB(38)"^":A=A+(P=3)-(P=2)
:A$=MID$("[s G][s M][s N][s H]",P,1):GOT
0
11 REM
12 REM SOLUTION TO PROBLEM #18-3 :
13 REM                 SKI SCROLL
14 REM     BY PAUL T. DAWSON
15 REM
```

The program suggested by Larry Snyder (Rapid City, SD) does not need the second line to be "scrunched." Use the two cursor keys to move the skier between the trees.

```
0 POKE53281,1:PRINT"[CLEAR][GREEN]":T=55
792:S=1520:X=500:FORI=0TO500:IFPEEK(S+J+
40)=65THENX=X-40
1 A=RND(1)*41:PRINTSPC(A)"[s A]":K=PEEK(
197):J=J+(K=7)-(K=2):POKES+J,22:POKET+J,
0:NEXT:PRINT"SCORE="X
11 REM
12 REM SOLUTION TO PROBLEM #18-3 :
13 REM                 SKI SCROLL
14 REM BY LARRY SNYDER
15 REM
```

Perhaps these programs will give some of you a starting point for creating an expanded version of the game. Think what you could do with a four or even a five line program!

Finally we have the solution to *Problem #18-4: Letter Math* from Donald Maloney (Manchester, CT). His program is representative of most of the others we received.

```
2 REM
3 REM SOLUTION TO PROBLEM #18-4 :
4 REM                LETTER MATH
5 REM BY DON MALONEY
6 REM
10 FOR X=0TO9:FOR Y=0TO9:FOR Z=1TO9
20 IF X+X*10+Y*100=X*100+Y*10+Z+Z*10+Y+X
 THEN PRINT"X, Y, Z = ";X;Y;Z
30 NEXT Z : NEXT Y : NEXT X
```

Some readers simplified the formulas in line 20 before programming them. You might try solving other similar problems using Donald's program as an example if you had trouble with this one. Readers with solutions to this problem who haven't already been mentioned include Chip Jones (Texarkana, AR), Jim Barranti (Canonsburg, PA) and Robert G. Bailey (Rockford, IL).

Thanks also to Ted Grondski (Springfield, MA), Alain Goulet (Port Colborne, ONT), and Greg Smisek (Lonsdale, MN) for their programs. One final teaser: What is the longest one-liner (total number of characters when listed) that you can come up with? Keep those problems and solutions coming. ☐

PROGRAM LISTINGS

Attention new *Ahoy!* readers! You must read the following information very carefully prior to typing in programs listed in *Ahoy!* Certain Commodore characters, commands, and strings of characters and commands will appear in a special format. Follow the instructions and listing guide on this page.

On the following pages you'll find several programs that you can enter on your Commodore computer. But before doing so, read this entire page carefully.

To insure clear reproductions, *Ahoy!*'s program listings are generated on a daisy wheel printer, incapable of printing the commands and graphic characters used in Commodore programs. These are therefore represented by various codes enclosed in brackets [ ]. For example: the SHIFT CLR/HOME command is represented onscreen by a heart . The code we use in our listings is [CLEAR]. The chart below lists all such codes which you'll encounter in our listings, except for one other special case.

The other special case is the COMMODORE and SHIFT characters. On the front of most keys are two symbols. The symbol on the left is obtained by pressing that key while holding down the COMMODORE key; the symbol on the right, by pressing that key while holding down the SHIFT key. COMMODORE and SHIFT characters are represented in our listings by a lower-case "s" or "c" followed by the symbol of the key you must hit. COMMODORE J, for example, is represented by [c J], and SHIFT J by [s J].

Additionally, any character that occurs more than two times in a row will be displayed by a coded listing. For example, [3 "[LEFT]"] would be 3 CuRSoR left commands in a row, [5 "[s EP]"] would be 5 SHIFTed English Pounds, and so on. Multiple blank spaces will be noted in similar fashion: e.g., 22 spaces as [22 " "].

Sometimes you'll find a program line that's too long for the computer to accept (C-64 lines are a maximum of 80 characters, or 2 screen lines long; VIC 20 lines, a maximum of 88 characters, or 4 screen lines). To enter these lines, refer to the *BASIC Command Abbreviations Appendix* in your User Manual.

On the next page you'll find our *Bug Repellent* programs for the VIC 20 and C-64. The version appropriate for your machine will help you proofread our programs after you type them. (Please note: the *Bug Repellent* line codes that follow each program line, in the whited-out area, should *not* be typed in. See the instructions preceding each program.)

Also on the following page you will find *Flankspeed*, our ML entry program, and instructions on its use. □

Call *Ahoy!* at 212-239-0855 with any problems.

When You See	It Means	You Type	You Will See
[CLEAR]	Screen Clear	SHIFT CLR/HOME	
[HOME]	Home	CLR/HOME	
[UP]	Cursor Up	SHIFT ↑ CRSR ↓	
[DOWN]	Cursor Down	↑ CRSR ↓	
[LEFT]	Cursor Left	SHIFT ←CRSR→	
[RIGHT]	Cursor Right	←CRSR→	
[SS]	Shifted Space	SHIFT Space	
[INSERT]	Insert	SHIFT INST/DEL	
[DEL]	Delete	INST/DEL	
[RVSON]	Reverse On	CNTRL 9	
[RVSOFF]	Reverse Off	CNTRL 0	
[UPARROW]	Up Arrow	↑	
[BACKARROW]	Back Arrow	←	
[PI]	PI	π	
[EP]	English Pound	£	

When You See	It Means	You Type	You Will See
[BLACK]	Black	CNTRL 1	
[WHITE]	White	CNTRL 2	
[RED]	Red	CNTRL 3	
[CYAN]	Cyan	CNTRL 4	
[PURPLE]	Purple	CNTRL 5	
[GREEN]	Green	CNTRL 6	
[BLUE]	Blue	CNTRL 7	
[YELLOW]	Yellow	CNTRL 8	
[F1]	Function 1	F1	
[F2]	Function 2	SHIFT F1	
[F3]	Function 3	F3	
[F4]	Function 4	SHIFT F3	
[F5]	Function 5	F5	
[F6]	Function 6	SHIFT F5	
[F7]	Function 7	F7	
[F8]	Function 8	SHIFT F7	

IMPORTANT! Letters on white background are **Bug Repellent** line codes. **Do not enter them!** This and the preceding explain these codes and provide other essential information on entering **Ahoy!** programs. Read these pages **before** entering programs.

BUG REPELLENT

This program will let you debug any *Ahoy!* program. Follow instructions for VIC 20 (cassette or disk) or C-64.

VIC 20 VERSION

By Michael Kleinert and David Barron

For cassette: type in and save the *Bug Repellent* program, then type RUN 63000[RETURN]SYS 828[RETURN]. If you typed the program properly, it will generate a set of two-letter line codes that will match those listed to the right of the respective program lines.

Once you've got a working *Bug Repellent*, type in the program you wish to check. Save it and type the RUN and SYS commands listed above once again, then compare the line codes generated to those listed in the magazine. If you spot a discrepancy, a typing error exists in that line. Important: you must use exactly the same spacing as the program in the magazine. Due to memory limitations on the VIC, the VIC *Bug Repellent* will register an error if your spacing varies from what's printed.

You may type SYS 828 as many times as you wish, but if you use the cassette for anything, type RUN 63000 to restore the *Repellent*.

When your program has been disinfected you may delete all lines from 63000 on. (Be sure the program you type doesn't include lines above 63000!)

For disk: enter *Bug Repellent*, save it, and type RUN:NEW [RETURN]. Type in the program you wish to check, then SYS 828.

To pause the line codes listing, press SHIFT.

To send the list to the printer type OPEN 4,4:CMD 4:SYS 828[RETURN]. When the cursor comes back, type PRINT#4:CLOSE 4[RETURN].

```
•63000 FORX=828TO1023:READY:POKEX,Y:NEXT:END   AC
•63001 DATA169,0,133,63,133,64,165,43,133,251  JL
•63002 DATA165,44,133,252,160,0,132,254,32,228 DF
•63003 DATA3,234,177,251,208,3,76,208,3,230    OE
•63004 DATA251,208,2,230,252,169,244,160,3,32  OH
•63005 DATA30,203,160,0,177,251,170,230,251,20
 8                                             KO
•63006 DATA2,230,252,177,251,32,205,221,169,58 JJ
•63007 DATA32,210,255,169,0,133,253,230,254,32 OK
•63008 DATA228,3,234,165,253,160,0,170,177,251 LG
•63009 DATA201,32,240,6,138,113,251,69,254,170 BP
•63010 DATA138,133,253,177,251,208,226,165,253
 ,41                                           DD
•63011 DATA240,74,74,74,74,24,105,65,32,210    EK
•63012 DATA255,165,253,41,15,24,105,65,32,210  FO
•63013 DATA255,169,13,32,210,255,173,141,2,41  PK
•63014 DATA1,208,249,230,63,208,2,230,64,230   CB
•63015 DATA251,208,2,230,252,76,74,3,169,236   KH
•63016 DATA160,3,32,30,203,166,63,165,64,32    DP
•63017 DATA205,221,169,13,32,210,255,96,230,25
 1                                             EL
•63018 DATA208,2,230,252,96,0,76,73,78,69      OI
•63019 DATA83,58,32,0,76,73,78,69,32,35        FG
•63020 DATA32,0,0,0,0,0                        LE
```

C-64 VERSION

By Michael Kleinert and David Barron

Type in, SAVE, and RUN the *Bug Repellent*. Type NEW, then type in or LOAD the *Ahoy!* program you wish to check. When that's done, SAVE your program (don't RUN it!) and type SYS 49152 [RETURN].

To pause the listing depress and hold the SHIFT key.

Compare the codes your machine generates to the codes listed to the right of the respective program lines. If you spot a difference, an error exists in that line. Jot down the number of lines where contradictions occur, LIST each line, spot the errors, and correct them.

```
•5000 FORX=49152TO49488:READY:POKEX,Y:NEXT:END GJ
•5001 DATA32,161,192,165,43,133,251,165,44,133 DL
•5002 DATA252,160,0,132,254,32,37,193,234,177  DB
•5003 DATA251,208,3,76,138,192,230,251,208,2   OF
•5004 DATA230,252,76,43,192,76,73,78,69,32     KN
•5005 DATA35,32,0,169,35,160,192,32,30,171     CA
•5006 DATA160,0,177,251,170,230,251,208,2,230  CE
•5007 DATA252,177,251,32,205,189,169,58,32,210 JE
•5008 DATA255,169,0,133,253,230,254,32,37,193  CL
•5009 DATA234,165,253,160,0,76,13,193,133,253  NB
•5010 DATA177,251,208,237,165,253,41,240,74,74 MB
•5011 DATA74,74,24,105,65,32,210,255,165,253   EP
•5012 DATA41,15,24,105,65,32,210,255,169,13    GH
•5013 DATA32,220,192,230,63,208,2,230,64,230   AN
•5014 DATA251,208,2,230,252,76,11,192,169,153  NG
•5015 DATA160,192,32,30,171,166,63,165,64,76   BF
•5016 DATA231,192,96,76,73,78,69,83,58,32      EP
•5017 DATA0,169,247,160,192,32,30,171,169,3    PJ
•5018 DATA133,254,32,228,255,201,83,240,6,201  FK
•5019 DATA80,208,245,230,254,32,210,255,169,4  FL
•5020 DATA166,254,160,255,32,186,255,169,0,133 CL
•5021 DATA63,133,64,133,2,32,189,255,32,192    GC
•5022 DATA255,166,254,32,201,255,76,73,193,96  NN
•5023 DATA32,210,255,173,141,2,41,1,208,249    NH
•5024 DATA96,32,205,189,169,13,32,210,255,32   IM
•5025 DATA204,255,169,4,76,195,255,147,83,67   KC
•5026 DATA82,69,69,78,32,79,82,32,80,82        DC
•5027 DATA73,78,84,69,82,32,63,32,0,76         ML
•5028 DATA44,193,234,177,251,201,32,240,6,138  GN
•5029 DATA113,251,69,254,170,138,76,88,192,0   JK
•5030 DATA0,0,0,230,251,208,2,230,252,96       NA
•5031 DATA170,177,251,201,34,208,6,165,2,73    DM
•5032 DATA255,133,2,165,2,208,218,177,251,201  JA
•5033 DATA32,208,212,198,254,76,29,193,0,169   FM
•5034 DATA13,76,210,255,0,0,0                  PA
```

FLANKSPEED FOR THE C-64

By Gordon F. Wheat

Flankspeed will allow you to enter machine language *Ahoy!* programs without any mistakes. Once you have typed the program in, save it for future use. While entering an ML program with *Flankspeed* there is no need to enter spaces or hit the carriage return. This is all done automatically. If you make an error in a line a bell will ring and you will be asked to enter it again. To LOAD in a program Saved with *Flankspeed* use LOAD "name",1,1 for tape, or LOAD "name",8,1 for disk. The function keys may be used after the starting and ending addresses have been entered.

f1 – SAVEs what you have entered so far.
f3 – LOADs in a program worked on previously.
f5 – To continue on a line you stopped on after LOADing in the previously saved work.
f7 – Scans through the program to locate a particular line, or to find out where you stopped the last time you entered the program. f7 temporarily freezes the output as well.

```
•5 POKE53280,12:POKE53281,11                   LL
•6 PRINT"[CLEAR][c 8][RVSON][15" "]FLANKSPEED[
 15" "]";                                      ED
•10 PRINT"[RVSON][5" "]MISTAKEPROOF ML ENTRY P
 ROGRAM[6" "]"                                 MC
•15 PRINT"[RVSON][9" "]CREATED BY G. F. WHEAT[
 9" "]"                                        DM
•20 PRINT"[RVSON][3" "]COPR. 1984, ION INTERNA
```

```
TIONAL INC.[3" "]"                          DH
·30 FORA=54272TO54296:POKEA,0:NEXT          IM
·40 POKE54272,4:POKE54273,48:POKE54277,0:POKE5
4278,249:POKE54296,15                       NH
·70 FORA=680TO699:READB:POKEA,B:NEXT        KO
·75 DATA169,251,166,253,164,254,32,216,255,96 HJ
·76 DATA169,0,166,251,164,252,32,213,255,96 JB
·80 B$="STARTING ADDRESS IN HEX":GOSUB2010:AD=
B:SR=B                                      HC
·85 GOSUB2520:IFB=0THEN80                   FO
·86 POKE251,T(4)+T(3)*16:POKE252,T(2)+T(1)*16 KE
·90 B$="ENDING ADDRESS IN HEX":GOSUB2010:EN=B IF
·95 GOSUB2510:IFB=0THEN80                   FP
·96 POKE254,T(2)+T(1)*16:B=T(4)+1+T(3)*16   MN
·97 IFB>255THENB=B-255:POKE254,PEEK(254)+1  GE
·98 POKE253,B:PRINT                         HN
·100 REM  GET HEX LINE                      IL
·110 GOSUB3010:PRINT": [c P][LEFT]";:FORA=0TO8 FG
·120 FORB=0TO1:GOTO210                      MD
·125 NEXTB                                  ME
·130 A%(A)=T(1)+T(0)*16:IFAD+A-1=ENTHEN310  LH
·135 PRINT" [c P][LEFT]";                   IK
·140 NEXTA:T=AD-(INT(AD/256)*256):PRINT" "  PD
·150 FORA=0TO7:T=T+A%(A):IFT>255THENT=T-255 LK
·160 NEXT                                   IA
·170 IFA%(8)<>TTHENGOSUB1010:GOTO110        FK
·180 FORA=0TO7:POKEAD+A,A%(A):NEXT:AD=AD+8:GOT
O110                                        MN
·200 REM    GET HEX INPUT                   AB
·210 GETA$:IFA$=""THEN210                   HO
·211 IFA$=CHR$(20)THEN270                   GC
·212 IFA$=CHR$(133)THEN4000                 MD
·213 IFA$=CHR$(134)THEN4100                 KF
·214 IFA$=CHR$(135)THENPRINT" ":GOTO4500    GE
·215 IFA$=CHR$(136)THENPRINT" ":GOTO4700    BJ
·220 IFA$>"@"ANDA$<"G"THENT(B)=ASC(A$)-55:GOTO
250                                         GM
·230 IFA$>"/"ANDA$<":"THENT(B)=ASC(A$)-48:GOTO
250                                         LE
·240 GOSUB1100:GOTO210                      LL
·250 PRINTA$"[c P][LEFT]";                  OA
·260 GOTO125                                CG
·270 IFA>0THEN280                           OP
·272 A=-1:IFB=1THEN290                      OB
·274 GOTO140                                CJ
·280 IFB=0THENPRINTCHR$(20);CHR$(20);:A=A-1 HG
·285 A=A-1                                  BE
·290 PRINTCHR$(20);:GOTO140                 KH
·300 REM LAST LINE                          AD
·310 PRINT" ":T=AD-(INT(AD/256)*256)        GJ
·320 FORB=0TOA-1:T=T+A%(B):IFT>255THENT=T-255 PL
·330 NEXT                                   IA
·340 IFA%(A)<>TTHENGOSUB1010:GOTO110        KF
·350 FORB=0TOA-1:POKEAD+B,A%(B):NEXT        HN
·360 PRINT:PRINT"YOU ARE FINISHED!":GOTO4000 ON
·1000 REM BELL AND ERROR MESSAGES           FL
·1010 PRINT:PRINT"LINE ENTERED INCORRECTLY":PR
INT:GOTO1100                                DH
·1020 PRINT:PRINT"INPUT A 4 DIGIT HEX VALUE!":
GOTO1100                                    JA
·1030 PRINT:PRINT"ENDING IS LESS THAN STARTING
!":B=0:GOTO1100                             HD
·1040 PRINT:PRINT"ADDRESS NOT WITHIN SPECIFIED
 RANGE!":B=0:GOTO1100                       AG
·1050 PRINT:PRINT"NOT ZERO PAGE OR ROM!":B=0:G
OTO1100                                     KN
·1060 PRINT"?ERROR IN SAVE":GOTO1100        EI
·1070 PRINT"?ERROR IN LOAD":GOTO1100        GL
·1080 PRINT:PRINT:PRINT"END OF ML AREA":PRINT PG
·1100 POKE54276,17:POKE54276,16:RETURN      BH
·1200 OPEN15,8,15:INPUT#15,A,A$:CLOSE15:PRINTA
$:RETURN                                    IM
·2000 REM GET FOUR DIGIT HEX                PC
·2010 PRINT:PRINTB$;:INPUTT$                GM
·2020 IFLEN(T$)<>4THENGOSUB1020:GOTO2010    II
·2040 FORA=1TO4:A$=MID$(T$,A,1):GOSUB2060:IFT(
A)=16THENGOSUB1020:GOTO2010                 AD
·2050 NEXT:B=(T(1)*4096)+(T(2)*256)+(T(3)*16)+
T(4):RETURN                                 GF
·2060 IFA$>"@"ANDA$<"G"THENT(A)=ASC(A$)-55:RET
URN                                         EH
·2070 IFA$>"/"ANDA$<":"THENT(A)=ASC(A$)-48:RET
URN                                         KP
·2080 T(A)=16:RETURN                        NP
·2500 REM ADRESS CHECK                      LI
·2510 IFAD>ENTHEN1030                       MI
·2515 IFB<SRORB>ENTHEN1040                  MG
·2520 IFB<256OR(B>40960ANDB<49152)ORB>53247THE
N1050                                       MI
·2530 RETURN                                IM
·3000 REM ADDRESS TO HEX                    EB
·3010 AC=AD:A=4096:GOSUB3070                HG
·3020 A=256:GOSUB3070                       CE
·3030 A=16:GOSUB3070                        PN
·3040 A=1:GOSUB3070                         MJ
·3060 RETURN                                IM
·3070 T=INT(AC/A):IFT>9THENA$=CHR$(T+55):GOTO3
090                                         CJ
·3080 A$=CHR$(T+48)                         JP
·3090 PRINTA$;:AC=AC-A*T:RETURN             AC
·4000 A$="**SAVE**":GOSUB4200               AI
·4050 OPEN1,T,1,A$:SYS680:CLOSE1            LH
·4060 IFST=0THENEND                         EO
·4070 GOSUB1060:IFT=8THENGOSUB1200          FJ
·4080 GOTO4000                              FF
·4100 A$="**LOAD**":GOSUB4200               AB
·4150 OPEN1,T,0,A$:SYS690:CLOSE1            MF
·4160 IFST=64THEN110                        JH
·4170 GOSUB1070:IFT=8THENGOSUB1200          CM
·4180 GOTO4100                              FO
·4200 PRINT" ":PRINTTAB(14)A$               FG
·4210 PRINT:A$="":INPUT"FILENAME";A$        OM
·4215 IFA$=""THEN4210                       GF
·4220 PRINT:PRINT"TAPE OR DISK?":PRINT      DF
·4230 GETB$:T=1:IFB$="D"THENT=8:A$="@0:"+A$:RE
TURN                                        IG
·4240 IFB$<>"T"THEN4230                     FN
·4250 RETURN                                IM
·4500 B$="CONTINUE FROM ADDRESS":GOSUB2010:AD=
B                                           DK
·4510 GOSUB2515:IFB=0THEN4500               MA
·4520 PRINT:GOTO110                         OI
·4700 B$="BEGIN SCAN AT ADDRESS":GOSUB2010:AD=
B                                           FH
·4705 GOSUB2515:IFB=0THEN4700               NK
·4706 PRINT:GOTO4740                        DI
·4710 FORB=0TO7:AC=PEEK(AD+B):GOSUB3030:IFAD+B
=ENTHENAD=SR:GOSUB1080:GOTO110              BK
·4715 PRINT" ";:NEXTB                       EC
·4720 PRINT:AD=AD+8                         GN
·4730 GETB$:IFB$=CHR$(136)THEN110           MN
·4740 GOSUB3010:PRINT": ";:GOTO4710         JD
```

IMPORTANT! Letters on white background are **Bug Repellent** line codes. **Do not enter them!** Pages 117 and 118 explain these codes and provide other essential information on entering *Ahoy!* programs. Refer to these pages **before** entering any programs!

INFINITESIMAL INTRIGUE

FROM PAGE 51 MOLECULAR MOTION

```
·1 REM                                          JD
·2 REM -- MOLECULAR MOTION --                   LI
·3 REM    RUPERT REPORT #22                     OL
·4 REM                                          JD
·6 REM MOVEMENT OF MOLECULES FROM ONE           OH
·7 REM COMPARTMENT TO ANOTHER                   GJ
·8 REM                                          JD
·9 :REM  TOTAL # AND # ON LEFT SIDE:            JA
·10 TTL=480 : NL=480                            II
·15 L1$=CHR$(157):L$=" "+L1$+L1$+" "            CG
·20 KK=1 : K1=1024 : K2=40                      JB
·30 K3=32 : K4=81  :REM BLANK & MOLECULE        JB
·35 :REM  CLEAR SCREEN AND SET COLOR:           LE
·40 POKE 53281,0 : PRINT CHR$(147)              HM
·50 POKE 53281,1 : PRINT CHR$(144)              GM
·55 :REM  PUT 480 MOLECULES ON LEFT:            GA
·60 FOR Y=1 TO 24 : FOR X=0 TO 19               PH
·70 POKE K1+X+K2*Y,K4 : NEXT : NEXT             KC
·75 :REM  ====  MAIN LOOP ====                  BA
·80 RL=NL/TTL :REM RELATIVE # ON LEFT           LP
·90 RN=RND(0) :REM RANDOM NUMBER                PP
·100 IF NL=480 AND KK=-1 THEN 270               KI
·110 IF RN<RL THEN PL=K3: PR=K4:GOTO 130        II
·120 PL=K4 : PR=K3                              NA
·130 X=INT(RND(0)*20):Y=INT(RND(0)*24)+1        JJ
·135 REM   MEMORY LOCATIONS TO POKE             KK
·140 ML=K1+X+Y*K2                               PK
·150 MR=K1+(20+X)+Y*K2                          HI
·160 GET A$ : IF A$<>"" THEN 270                DB
·165 REM  FIND LOCATION TO 'UNPOKE'             KL
·170 IF PEEK(ML)<>PL THEN 210                   HL
·180 Y=Y+1 : IF Y<25 THEN 140                   LE
·190 Y=1 : X=X+1 : IF X<20 THEN 140             GM
·200 X=0 : GOTO 140                             JN
·210 IF PL=K3 THEN NL=NL-KK : GOTO 230          FG
·220 NL=NL+KK                                   BN
·230 NR=TTL-NL : T=T+1                          ON
·240 POKE ML,PL : POKE MR,PR                    FC
·250 PRINT CHR$(19);NL;TAB(17);"T=";KK*T;
 L$;TAB(34);NR;L$                               AL
·260 GOTO 80                                    PA
·265 REM   REVERSE THE ARROW OF TIME            LF
·270 KK=-KK : TEMP=K3 : K3=K4 : K4=TEMP         OO
·280 T=0 : GOTO 80                              GM
```

RADIOACTIVE DECAY

```
·1 REM                                          JD
·2 REM  * RADIOACTIVE DECAY *                   FF
·3 REM    RUPERT REPORT #22                     OL
·4 REM                                          JD
·5 REM MONTE CARLO SIMULATION OF A              MN
·6 REM TWO-PRODUCT DECAY CHAIN                  NE
·7 REM                                          JD
·10 TTL=100    :REM INITIAL NUMBER              BJ
·20 SF=22/TTL :REM PLOT SCALE FACTOR            JH
·30 ND=2   :REM NUMBER OF DAUGHTERS             HL
·40 DIM NUC(TTL)                                LL
·100 REM  DECAY CONSTANTS                       KH
·110 DC(0)=.2                                   CB
·120 DC(1)=.05 :REM FIRST DAUGHTER              AO
·130 DC(2)=0    :REM STABLE ISOTOPE             IN
·140 M(0)=TTL  :REM INITIAL AMOUNT              JM
·150 PRINT CHR$(147)"  =-= RADIOACTIVE DE
 CAY =-="                                       LO
·160 FOR T=1 TO 20                              MF
·170 GOSUB 400 :REM <<< GOSUB 500 TO PLOT       NM
·180 FOR N=1 TO TTL                             FL
·190 IF RND(0) > DC(NUC(N)) THEN 220            GN
·200 K=NUC(N) : NUC(N)=K+1                      AG
·210 M(K)=M(K)-1 : M(K+1)=M(K+1)+1              MC
·220 NEXT N                                     NA
·230 NEXT T                                     NG
·240 POKE 214,23 : PRINT                        KE
·250 POKE 211,12                                LB
·260 PRINT"=PRESS A KEY TO CONTINUE==>";        NF
·270 GET A$ : IF A$="" THEN 270                 HE
·280 GOTO 150                                   CG
·390 END                                        IC
·395 REM --- PRINT VALUE ROUTINE ---            AA
·400 PRINT TM ">>",                             DN
·410 FOR J=0 TO ND                              OO
·420 PRINT M(J),:NEXT J                         LN
·430 TM=TM+1                                    MK
·440 RETURN                                     IM
·495 REM                                        JD
·499 REM ---- PLOT ROUTINE ----                 KB
·500 FOR J=0 TO ND                              OO
·510 Y=M(J) : S$=RIGHT$(STR$(J),1)              FN
·520 ROW%=22-SF*Y : COL%=2*T-2                  HL
·530 IF ROW%>23 THEN ROW%=23                    JG
·540 IF ROW%<=0 THEN ROW%=0                     ME
·550 IF COL%>38 THEN COL%=38                    IG
·560 IF COL%<0 THEN COL%=0                      PJ
·570 POKE 214,ROW%: PRINT                       GL
·580 POKE 211,COL%: PRINT S$                    IO
·590 NEXT J : RETURN                            JJ
```

ALIVE & IN LIVING COLOR

FROM PAGE 18 DRAGON

```
·1 REM     *** DRAGON ***                       EK
·2 REM MULTI-COLOR SPRITE ANIMATION             GE
·4 REM ** MOVE TOP OF MEMORY TO MAKE ROOM
 FOR VIDEO BLOCK AT 32768                       CA
```

```
•5 POKE 55,255:POKE 56,127:POKE 643,255:P
OKE 644,127                                KG
•9 REM      *** SET-UP SUBROUTINES ***     OF
•10 GOSUB 600:REM (VIDEO MEMORY)           II
•12 GOSUB 1000:REM (SPRITE SHAPES)         CB
•13 GOSUB 900:REM (SPRITE POSITIONS)       IE
•15 GOTO 100                               CF
•98 REM      *** ACTION LOOP ***           KJ
•100 FOR AN=0 TO 3                         MA
•105 K=KP(PEEK(653)):IF K>2 THEN 370       JK
•110 IF (K>1)AND(Q=1) THEN AN=0:Q=0:POKE
LT(0),ST(XK,AN):GOTO 105                   BM
•115 IF K>1 THEN 105                       PO
•120 Q=1:HP=HP+K(K):IF HP<0 THEN HP=HP+1   DA
•125 IF HP>HZ THEN HP=HP-1                 PP
•130 IF K<>XK THEN XK=K                    DN
•135 POKE HT(0),HH(HP):POKE HR,HB(HP)      AO
•140 POKE LT(0),ST(XK,AN)                  DD
•196 NEXT:GOTO 100                         NM
•296 RETURN                                IM
•369 REM ** PUT VIDEO MEMORY BACK TO FIRS
T BLOCK, AND SCREEN MEMORY TO 1024         EG
•370 POKE 56578,PEEK(56578)OR3:POKE 56576
,(PEEK(56576)AND 252)OR 3                  MB
•371 POKE 53272,20:POKE 648,4              OM
•379 REM ** REENABLE SHIFT/COMMODORE AND
RUN-STOP/RESTORE                           FB
•380 POKE 657,0:POKE 792,71:POKE 808,237   PK
•390 END                                   IC
•598 REM      *** ARRANGE MEMORY ***       KB
•599 REM USE THIRD VIDEO BLOCK (32768 TO
49151), SO ROM CHARACTER SET IS USABLE.    AL
•600 VB=32768:POKE 56578,PEEK(56578)OR3:P
OKE 56576,(PEEK(56576)AND 252)OR 1         CN
•601 REM ** TELL VIC-2 WHERE SCREEN IS WI
THOUT CHANGING CHARACTER SET LOCATION      GA
•602 SB=0:POKE 53272,(SB*16)+4:SB=VB+1024
*SB                                        MO
•603 REM ** TELL BASIC WHERE SCREEN IS     GB
•604 BB=SB/256:POKE 648,BB                 AK
•608 REM      *** REGISTER ADDRESSES ***   EJ
•609 REM ** SPRITE SHAPE LOCATION TABLE    IF
•610 LT(0)=SB+1016:FOR I=1 TO 7:LT(I)=LT(
I-1)+1:NEXT                                AF
•611 REM ** SPRITE COLOR TABLE             LA
•612 CT(0)=53287:FOR I=1 TO 7:CT(I)=CT(I-
1)+1:NEXT                                  AN
•613 REM ** SPRITE HORIZONTAL POSITION TA
BLE (LOW BYTES)                            CH
•614 HT(0)=53248:FOR I=1 TO 7:HT(I)=HT(I-
1)+2:NEXT                                  KC
•615 REM ** SPRITE VERTICAL POSITION TABL
E                                          IJ
•616 VT(0)=53249:FOR I=1 TO 7:VT(I)=VT(I-
1)+2:NEXT                                  AJ
•617 REM ** SPRITE HORIZONTAL HIGH-BIT RE
GISTER                                     IG
•618 HR=53264                              FB
•619 REM ** SPRITE ENABLE REGISTER         AF
•620 ES=53269                              FE
•621 REM ** VERTICAL EXPANSION REGISTER (
1=DOUBLE HEIGHT)                           IB
•622 VE=53271                              GC
•623 REM ** HORIZONTAL EXPANSION REGISTER
 (1=DOUBLE WIDTH)                          PL
•624 HE=53277                              FC
•625 REM ** SPRITE PRIORITY REGISTER (1=S
PRITE IS IN FRONT OF FOREGROUND)           GM
•626 PR=53275                              FJ
•627 REM ** MULTICOLOR ENABLE REGISTER (1
=MULTI-COLOR ENABLED)                      IM
•628 EM=53276                              EK
•629 REM ** SPRITE MULTICOLOR COLOR REGIS
TERS                                       OH
•630 MR=53285:REM ('01' REGISTER: ADD 1 T
O MR FOR '11' REGISTER)                    LJ
•631 REM ** SPRITE-SPRITE COLLISION REG:S
PRITE-FOREGROUND COLLISION REG             ID
•632 CS=53278:CF=53279                     AO
•633 REM ** SET-BIT AND CLEAR-BIT VALUES   BI
•634 BS(0)=1:FOR I=1 TO 7:BS(I)=2*BS(I-1)
:NEXT                                      NE
•635 FOR I=0 TO 7:BC(I)=255-BS(I):NEXT     LB
•636 REM ** MULTICOLOR SET-BIT VALUES      MN
•637 FORI=0TO3:MS(0,I)=I:NEXT:FORI=1TO3:F
ORJ=0TO3:MS(I,J)=MS(I-1,J)*4:NEXT:NEXT     OH
•638 REM      *** INITIALIZE VALUES ***    DK
•639 REM ** FOREGROUND COLOR               FL
•640 POKE 53281,9:PRINT "[CLEAR][c 1]";:R
EM (BROWN)                                 GG
•641 REM ** BACKGROUND COLOR               DK
•642 POKE 53281,0 :REM (BLACK)             JB
•643 REM ** BORDER COLOR                   BE
•644 POKE 53280,9:REM (BROWN)              LM
•645 REM ** SPRITE COLORS (DEFAULTS: WHI,
RED,L-GRN,PUR,GRN,BLU,YEL,M-GRAY)          IE
•646 POKE CT(0),5:REM (THE SCALES ARE GRE
EN)                                        HN
•647 REM ** SET PRIORITY                   EF
•648 POKE PR,0:REM (ALL IN FRONT)          OD
•649 REM ** SET HORIZONTAL SIZES           BJ
•650 POKE HE,1:REM (SPR 0=DOUBLE)          NJ
•651 REM ** SET VERTICAL SIZES             CC
•652 POKE VE,1:REM (SPR 0=DOUBLT)          MO
•653 REM ** ENABLE SPRITES                 OD
•654 POKE ES,BS(0):REM (SPRITE 0 ONLY)     KB
•655 REM ** ENABLE MULTICOLOR FOR SPR 0    DG
•656 POKE EM,BS(0)                         FN
•657 REM ** SET MULTI-COLORS 1 AND 3 (1=L
IGHT GREY, 3=FLAME YELLOW)                 DJ
•658 POKE MR,15:POKE MR+1,7                JA
•689 REM      *** SAFETY PROCEDURES ***    CM
•690 POKE 657,128:REM DISABLE SHIFT/COMMO
DORE CHARACTER SET SWITCH                  OA
•691 POKE 808,234:POKE 792,193:REM DISABL
E STOP AND STOP/RESTORE                    JJ
```

IMPORTANT! Letters on white background are **Bug Repellent** line codes. **Do not enter them!** Pages 117 and 118 explain these codes and provide other essential information on entering ***Ahoy!*** programs. Refer to these pages **before** entering any programs!

```
·696 RETURN                                  IM
·898 REM      *** SPRITE POSITIONS ***       KA
·899 REM ** POSSIBLE POSITIONS DIM'ED        MK
·900 VZ=19:HZ=102:DIM HH(HZ),VV(VZ),HB(HZ
)                                            NM
·901 REM ** ASSIGN  HORIZONTAL VALUES        IA
·902 X=8:B=0:FOR I=0 TO HZ:HH(I)=X:HB(I)=
B                                            PB
·903 X=X+3:IF X>255 THEN X=X-256:B=1         DC
·904 NEXT                                    IA
·905 REM ** ASSIGN VERTICAL VALUES           LF
·906 X=53:FOR I=0 TO VZ:VV(I)=X:X=X+8:NEX
T                                            JN
·919 REM ** KEYPRESS MATRIX                  EO
·920 DIM KP(7),K(2)                          JA
·921 FOR I=0 TO 7:KP(I)=2:NEXT:KP(1)=1:KP
(2)=0:KP(7)=3                                ML
·922 K(0)=-1:K(1)=1:K(2)=0                   BK
·929 REM ** INITIAL SPRITE POSITIONING       IO
·930 HP=6:VP=VZ:XK=1:AN=0                    PG
·931 POKE VT(0),VV(VP):POKE HT(0),HH(HP):
POKE HR,HB(HP)                               AC
·939 REM ** TELL VIC-2 WHERE TO FIND SPRI
TE SHAPE                                     HK
·940 POKE LT(0),ST(XK,AN)                    DD
·949 REM ** CLEAR SCREEN                     BI
·950 PRINT "[CLEAR]";                        KN
·996 RETURN                                  IM
·998 REM       *** SPRITE SHAPES ***         CD
·999 REM ** # SPR SHAPES -1; # SHAPES/ANI
MATION SEQ; # LINES/SHAPE -1                 OF
·1000 NS=1:AS=3:TS=11                        FJ
·1001 REM ** LOCATE SPRITE SHAPE MEMORY      OM
·1002 ST(0,0)=16:FOR I=1 TO NS:ST(I,0)=ST
(I-1,0)+AS+1:NEXT                            MP
·1003 FOR I=0 TO NS:FOR J=1 TO AS:ST(I,J)
=ST(I,J-1)+1:NEXT:NEXT                       GH
·1004 FOR I=0 TO NS:FOR J=0 TO AS:FOR K=0
 TO TS:READ SS$                              OL
·1005 FOR L=0 TO 2:X=0:Y=VB+ST(I,J)*64+K*
3+L                                          CC
·1006 REM ** CONVERT STRINGS TO SHAPE BYT
ES (USE MS(0-3,0-3) TO SET BITS)             MN
·1007 FOR M=1 TO 4:N=ASC(MID$(SS$,M+L*4,1
))                                           AJ
·1008 IF(N>48)AND(N<52)THEN X=X OR MS(4-M
,N-48)                                       AH
·1009 NEXT:POKE Y,X:NEXT:PRINT SS$:NEXT      GO
·1010 Z=64-(TS*3+2):FOR K=Y+1 TO Y+Z:POKE
 K,0:NEXT:NEXT:NEXT                          IE
·1017 REM ** SPRITE SHAPE DATA               HJ
·1018 REM .='OFF'   1=COLOR 1   2=COLOR 2
   3=COLOR 3                                 IO
·1019 REM ** 1ST SHAPE, FIRST SEQ            LG
·1020 DATA "[6"."]2[5"."]"                   BK
·1021 DATA "[4"."]22.2[4"."]"                AK
·1022 DATA "[4"."][4"2"][4"."]"              CO
·1023 DATA "[6"."]22[4"."]"                  DO
·1024 DATA "[5"."]1[3"2"][3"."]"             AF
·1025 DATA "[4"."]22122[3"."]"               AB
·1026 DATA "[5"."]1122.2."                   BI
·1027 DATA "[5"."]1122..2"                   AA
·1028 DATA "[6"."]122..2"                    BF
·1029 DATA "[7"."]22..2"                     BC
·1030 DATA "[6"."]1[3"2"].2"                 BJ
·1031 DATA "[6"."][3"2"].2."                 DO
·1039 REM ** 2ND SHAPE, 1ST SEQUENCE         FG
·1040 DATA "[6"."]2[5"."]"                   BK
·1041 DATA "[4"."]22.2[4"."]"                AK
·1042 DATA "[3"."][3"3"]22[4"."]"            AB
·1043 DATA "[4"."][4"2"][4"."]"              CO
·1044 DATA "[4"."]11[3"2"][3"."]"            DC
·1045 DATA "[5"."]2212[3"."]"                AP
·1046 DATA "[5"."][3"1"]2..2"                AJ
·1047 DATA "[5"."][3"1"]2..2"                AJ
·1048 DATA "[6"."]112.2."                    AK
·1049 DATA "[7"."]12.2."                     BD
·1050 DATA "[5"."][3"1"][3"2"]."             AF
·1051 DATA "[6"."][3"2"][3"."]"              AC
·1059 REM ** 3RD SHAPE, 1ST SEQUENCE         GL
·1060 DATA "[6"."]2[5"."]"                   BK
·1061 DATA "..3.22.2[4"."]"                  AL
·1062 DATA ".3.3..22[4"."]"                  AG
·1063 DATA "..3.[4"2"][4"."]"                CP
·1064 DATA "[5"."]1[3"2"][3"."]"             AF
·1065 DATA "[4"."]22122[3"."]"               AB
·1066 DATA "[5"."]1122..2"                   AA
·1067 DATA "[5"."]1122.2."                   BI
·1068 DATA "[6"."]122.2."                    AN
·1069 DATA "[7"."]22..2"                     BC
·1070 DATA "[6"."]1[5"2"]"                   BF
·1071 DATA "[6"."][3"2"][3"."]"              AC
·1079 REM ** 4TH SHAPE, 1ST SEQUENCE         LC
·1080 DATA "3[5"."]2[5"."]"                  BB
·1081 DATA ".3..22.2[4"."]"                  AD
·1082 DATA "3.3.[4"2"][4"."]"                CO
·1083 DATA ".3[4"."]22[4"."]"                BH
·1084 DATA "3[5"."][3"2"][3"."]"             AJ
·1085 DATA "[4"."][5"2"][3"."]"              DC
·1086 DATA "[5"."]1[3"2"].2."                AB
·1087 DATA "[5"."]1[3"2"].2."                AB
·1088 DATA "[6"."][3"2"]..2"                 AG
·1089 DATA "[7"."]22..2"                     BC
·1090 DATA "[7"."][3"2"].2"                  BG
·1091 DATA "[5"."][4"2"].2."                 AC
·1119 REM ** 1ST SHAPE, 2ND SEQUENCE         HE
·1120 DATA "[5"."]2[6"."]"                   BK
·1121 DATA "[4"."]2.22[4"."]"                DK
```

```
•1122 DATA "[4"."][4"2"][4"."]"                 CO
•1123 DATA "[4"."]22[6"."]"                     AG
•1124 DATA "[3"."][3"2"]1[5"."]"                BN
•1125 DATA "[3"."]22122[4"."]"                  AB
•1126 DATA ".2.2211[5"."]"                      DI
•1127 DATA "2..2211[5"."]"                      DI
•1128 DATA "2..221[6"."]"                       DN
•1129 DATA "2..22[7"."]"                        AK
•1130 DATA "2.[3"2"]1[6"."]"                    BJ
•1131 DATA ".2.[3"2"][6"."]"                    CO
•1139 REM ** 2ND SHAPE, 3RD SEQ                 PM
•1140 DATA "[5"."]2[6"."]"                      BK
•1141 DATA "[4"."]2.22[4"."]"                   DK
•1142 DATA "[4"."]22[3"3"][3"."]"               DF
•1143 DATA "[4"."][4"2"][4"."]"                 CO
•1144 DATA "[3"."][3"2"]11[4"."]"               DO
•1145 DATA "[3"."]2122[5"."]"                   OH
•1146 DATA "2..2[3"1"][5"."]"                   DB
•1147 DATA "2..2[3"1"][5"."]"                   DB
•1148 DATA ".2.211[6"."]"                       CO
•1149 DATA ".2.21[7"."]"                        AD
•1150 DATA ".[3"2"][3"1"][5"."]"                BN
•1151 DATA "[3"."][3"2"][6"."]"                 BK
•1160 DATA "[5"."]2[6"."]"                      BK
•1161 DATA "[4"."]2.22.3.."                     DL
•1162 DATA "[4"."]22..3.3."                     AG
•1163 DATA "[4"."][4"2"].3.."                   CP
•1164 DATA "[3"."][3"2"]1[5"."]"                BN
•1165 DATA "[3"."]22122[4"."]"                  AB
•1166 DATA "2..2211[5"."]"                      DI
•1167 DATA ".2.2211[5"."]"                      DI
•1168 DATA ".2.2211[5"."]"                      DI
•1169 DATA "2..22[7"."]"                        AK
•1170 DATA "[5"2"]1[6"."]"                      CN
•1171 DATA "[3"."][3"2"][6"."]"                 BK
•1179 REM ** 4TH SHAPE, 2ND SEQUENCE            HE
•1180 DATA "[5"."]2[5"."]3"                     DN
•1181 DATA "[4"."]2.22..3."                     DD
•1182 DATA "[4"."][4"2"].3.3"                   DG
•1183 DATA "[4"."]22[4"."]3."                   DP
•1184 DATA "[3"."][3"2"][5"."]3"                DN
•1185 DATA "[3"."][5"2"][4"."]"                 BC
•1186 DATA ".2.[3"2"]1[5"."]"                   DB
•1187 DATA ".2.[3"2"]1[5"."]"                   DB
•1188 DATA "2..[3"2"][6"."]"                    CO
•1189 DATA "2..22[7"."]"                        AK
•1190 DATA "2.22[8"."]"                         BK
•1191 DATA ".2.[4"2"][5"."]"                    DC
•1990 RETURN                                    IM
```

SOLITAIRE 64

FROM PAGE 37

```
•1 REM         SOLITAIRE 64                     KE
•2 REM        BY ERIC W SINK                    OM
•3 POKE55,255:POKE56,63:CLR:POKE53269,.:P
OKE53271,.:POKE53277,.                          EH
•4 X=RND(-TI)                                   FH
•5 S4=64:O4=14:T6=26:H9=39:Z1=1:Z2=2:Z3=3
:Z4=4:O3=13                                     LI
•6 POKE53280,0:POKE53281,4:PRINT"[CLEAR][
YELLOW]";CHR$(142)"[008]"                       KM
•7 IFPEEK(2)=15THENPRINTCHR$(14)"[008]"         AM
•8 GOSUB103                                     CK
•9 GOSUB54:GOSUB101:GOSUB72:GOSUB190:GOSU
B216                                            IO
•10 CLR:DIM P(7,20),H(13),C(13),D(13),S(1
3),NP(7),DK%(52)                                HF
•11 S4=64:O4=14:T6=26:H9=39:Z1=1:Z2=2:Z3=
3:Z4=4:O3=13                                    LI
•12 M$="HANG ON, STILL SETTING UP[3"."]":
P$="[HOME][DOWN][DOWN]":PRINT"[CLEAR]":G
OSUB108                                         OK
•13 X=RND(-TI)                                  FH
•14 DP=0:NH=0:NC=0:ND=0:NS=0:DN=0               AC
•15 PRINT:PRINT:PRINT"SHUFFLING[3"."]"          NA
•16 FORI=1TO52                                  LD
•17 CD=INT(RND(1)*52)+Z1:IFDK%(CD)THEN17        BJ
•18 DK%(CD)=I:NEXT                              FI
•19 PRINT"DEALING OUT CARDS[3"."]"              DN
•20 FORI=1TO28                                  MM
•21 F1=.:CD=DK%(53-I):DK%(53-I)=.               AG
•22 IFI=7ORI=13ORI=18ORI=22ORI=25ORI=27OR
I=28THENCD=CD+S4:F1=1                           CJ
•23 PI=PI+1:P(PI,NP(PI)+1)=CD:NP(PI)=NP(P
I)+1:IFF1=1THENPI=0                             DE
•24 NEXT                                        IA
•25 DN=24:DP=24:POKE53280,0:POKE53281,1         HE
•26 PRINT"[CLEAR][RVSON][BLUE][3" "]* * *
[3" "]S O L I T A I R E[3" "]* * *[3" "]
";:POKE16423,160                                AI
•27 GOSUB368                                    DF
•28 PRINT"[CLEAR][RVSON][BLUE][3" "]* * *
[3" "]S O L I T A I R E[3" "]* * *[3" "]
";:POKE16423,160                                AI
•29 IFLV=1THENGOSUB382                          OF
•30 POKE55335,11:PRINT:PRINT"[HOME]";           HF
•31 FORQ=1TO7:PN=Q:GOSUB111:NEXT                JH
•32 GOSUB155                                    DB
•33 GOSUB187:GOSUB226                           JE
•34 GOSUB243:REM FIRST MOVE                     PL
•35 GOSUB269:REM SECOND MOVE                    OL
•36 IFBU=1THENBU=.:GOTO34                       GJ
•37 IFDNTHEN34                                  CH
•38 FORI=1TO7:IFP(I,1)ANDS4=S4ORNP(I)=0TH
ENNEXT:GOTO40                                   CD
•39 GOTO34                                      PF
•40 FORI=1TO7:NP(I)=.:NEXT:NH=O3:NC=O3:ND
=O3:NS=O3                                       LP
•41 H(NH)=O3:C(NC)=O3:D(ND)=O3:S(NS)=O3         PF
•42 GOSUB155:GOSUB137:FORHI=1TO7:PN=HI:GO
SUB111:NEXT                                     BD
•43 IFLV=1THEN374                               GB
•44 PRINT"[HOME][6"[DOWN]"]"TAB(15);"[RVS
ON]YOU WON[3"!"]"                               MK
•45 PRINT"[HOME][9"[DOWN]"]"TAB(14);"[RVS
```

```
ON][RED]PLAY AGAIN??"                    AO
•46 PRINT:PRINT                          GJ
•47 PRINTTAB(9)"[RVSON]PRESS UP TO PLAY A
GAIN"                                    KO
•48 PRINT                                JJ
•49 PRINTTAB(11)"[RVSON]PRESS DOWN TO QUI
T[BLACK]"                                PN
•50 X=PEEK(56320):U=XAND1:V=XAND2:IFU=1AN
DV=2THEN50                               CP
•51 IFU=.THENRUN10                       FL
•52 PRINT"[CLEAR][BLACK]";:POKE53269,.:EN
D                                        FH
•53 REM                                  JD
•54 REM                                  JD
•55 POKE2,15                             GH
•56 DATA173,000,221,041,252,009,002      IL
•57 DATA141,000,221,169,002,141,024      GH
•58 DATA208,169,064,141,136,002,169      GC
•59 DATA147,032,210,255,169,127,141      NI
•60 DATA013,220,169,051,133,001,160      GF
•61 DATA000,169,000,133,251,133,253      BA
•62 DATA169,208,133,252,169,072,133      PI
•63 DATA254,177,251,145,253,200,192      JO
•64 DATA000,208,247,230,252,230,254      EN
•65 DATA165,252,201,224,208,237,169      AI
•66 DATA055,133,001,169,129,141,013      HI
•67 DATA220,096,000                      LB
•68 PRINT"[HOME]E"                       DK
•69 C1=.:FORI=49152TO49231:READB:C1=C1+B:
POKEI,B:POKE1024,B:POKE16384,B:NEXT      BI
•70 IFC1<>11494THENPRINT"ERROR IN DATA LI
NES 10010-10065":STOP                    MC
•71 RETURN                               IM
•72 REM                                  JD
•73 DATA064,007,012,024,024,024,024,024,0
24                                       DE
•74 DATA065,255,000,240,144,240,144,144,0
00                                       KE
•75 DATA066,255,000,240,016,240,128,240,0
00                                       EI
•76 DATA067,255,000,240,016,240,016,240,0
00                                       CH
•77 DATA068,255,000,144,144,240,016,016,0
00                                       AE
•78 DATA069,255,000,240,128,240,016,240,0
00                                       IN
•79 DATA070,255,000,240,128,240,144,240,0
00                                       PN
•80 DATA071,255,000,240,016,016,016,016,0
00                                       OM
•81 DATA072,255,000,240,144,240,144,240,0
00                                       AJ
•82 DATA073,255,000,240,144,240,016,240,0
00                                       AC
•83 DATA074,255,000,152,164,164,164,152,0
00                                       BM
•84 DATA075,255,000,112,032,032,160,224,0
00                                       JH
•85 DATA076,255,000,240,144,144,240,024,0
00                                       LN
•86 DATA077,255,000,144,160,192,160,144,0
00                                       GA
•87 DATA078,255,000,054,127,127,062,028,0
08                                       EG
•88 DATA079,255,000,012,012,051,051,012,0
30                                       CC
•89 DATA080,255,000,008,028,062,028,008,0
00                                       FA
•90 DATA081,255,000,024,060,126,126,024,0
60                                       NB
•91 DATA082,224,048,024,024,024,024,024,0
24                                       DM
•92 DATA085,024,024,024,024,024,024,012,0
07                                       FF
•93 DATA086,024,024,024,024,024,024,048,2
24                                       DI
•94 DATA087,024,024,024,024,024,024,024,0
24                                       JH
•95 DATA088,000,000,000,000,000,000,000,2
55                                       PI
•96 DATA-1                               OE
•97 PRINT"[CLEAR]E"                      MD
•98 READA:IFA=-1THENRETURN               KN
•99 POKE16384,A                          AF
•100 FORI=0TO7:READB:POKE18432+A*8+I,B:NE
XT:GOTO98                                MF
•101 REM                                 JD
•102 SYS49152:RETURN                     GI
•103 REM                                 JD
•104 M$="[15" "]SOLITAIRE":P$="[HOME][7"[
DOWN]"]":GOSUB108                        GJ
•105 M$="[13" "]BY ERIC W SINK":P$="[HOME
][11"[DOWN]"]":GOSUB108                  NE
•106 M$="[8" "]PLEASE WAIT[3"."]SETTING U
P":P$="[HOME][16"[DOWN]"]":GOSUB108      CH
•107 RETURN                              IM
•108 REM                                 JD
•109 FORI=1TOLEN(M$):PRINTP$RIGHT$(M$,I);
:NEXT                                    FP
•110 RETURN                              IM
•111 REM                                 JD
•112 REM                                 JD
•113 REM                                 JD
•114 REM                                 JD
•115 HP=3+(PN-1)*5:UC=.                  JD
•116 VP=5:ZQ=.                           HK
•117 PRINT"[HOME]";:FORI=1TOVP:PRINT"[DOW
N]";:NEXT:PRINTTAB(HP);                  OC
•118 IFNP(PN)THEN120                     FD
•119 PRINT"[4" "]":FORI=1TO13:PRINTTAB(HP
);"[4" "]":NEXT:GOTO135                  LD
•120 FORI=1TONP(PN):CD=P(PN,I):IFI=SXTHEN
ZQ=1                                     MD
•121 IFCD<S4THENUC=UC+Z1:GOTO130         DM
```

```
·122 CD=CD-S4:SU=Z4:IFCD<O4THENSU=Z1 JF
·123 IFCD>=O4ANDCD<=T6THENSU=Z2 ML
·124 IFCD>T6ANDCD<=H9THENSU=Z3 OO
·125 CH=CD-((SU-1)*13) OB
·126 SU$=CHR$(205+SU):CO$="[BLACK]":IFSU=
 1ORSU=3THENCO$="[RED]" AH
·127 IFZC=1ANDZQ=1THENCO$="[BLUE]" OF
·128 CD$=CO$+CHR$(192)+CHR$(192+CH)+SU$+C
 HR$(210) OO
·129 PRINTCD$ EH
·130 PRINTTAB(HP); DE
·131 NEXT IA
·132 PRINT"[s -]  [s -]":PRINTTAB(HP)"[s
 U][c @][c @][s V]" DG
·133 ZC=.:ZQ=. EA
·134 FORI=1TO12-NP(PN)+UC:PRINTTAB(HP);"[
 4" "]":NEXT CP
·135 PRINT"[HOME][BLUE]";:POKE214,3:PRINT
 :PRINTTAB(HP+1)UC BK
·136 RETURN IM
·137 REM ACES KP
·138 PRINT"[HOME][DOWN]"; DE
·139 IFNH=.THEN141 EI
·140 SU=Z1:CD=H(NH):HP=9:GOSUB151 NH
·141 PRINT"[HOME][DOWN]"; DE
·142 IFNC=.THEN144 EA
·143 SU=Z2:CD=C(NC):HP=15:GOSUB151 DG
·144 PRINT"[HOME][DOWN]"; DE
·145 IFND=.THEN147 DO
·146 SU=Z3:CD=D(ND):HP=21:GOSUB151 OK
·147 PRINT"[HOME][DOWN]"; DE
·148 IFNS=.THEN150 FD
·149 SU=Z4:CD=S(NS):HP=27:GOSUB151 JL
·150 RETURN IM
·151 REM ACE FN
·152 SU$=CHR$(205+SU):CO$="[BLACK]":IFSU=
 1ORSU=3THENCO$="[RED]" AH
·153 CD$=CO$+CHR$(192)+CHR$(192+CD)+SU$+C
 HR$(210) IK
·154 PRINTTAB(HP)CD$:PRINTTAB(HP)"[s -]
 [s -]":PRINTTAB(HP)"[s U][c @][c @][s V]
 ":RETURN HN
·155 REM DECK KG
·156 IFDPTHENGOSUB176:GOTO158 FE
·157 GOSUB180 DB
·158 PRINT"[HOME][20"[DOWN]"]"TAB(9); PK
·159 IFDP<>DNTHEN161 EF
·160 PRINT"[4" "]":PRINTTAB(9)"[4" "]":PR
 INTTAB(9)"[4" "]":PRINTTAB(9)"[4" "]":GO
 TO171 KB
·161 CD=DK%(DP+Z1):SU=4:IFCD<=O3THENSU=Z1 HJ
·162 IFCD>=O4ANDCD<=T6THENSU=Z2 ML
·163 IFCD>T6ANDCD<=H9THENSU=Z3 OO
·164 CH=CD-((SU-1)*O3) AH
·165 SU$=CHR$(205+SU):CO$="[BLACK]":IFSU=
 1ORSU=3THENCO$="[RED]" AH
·166 IFZC=1THENZC=.:CO$="[BLUE]" FL
·167 CD$=CO$+CHR$(192)+CHR$(192+CH)+SU$+C
 HR$(210) OO
·168 PRINTCD$ EH
·169 PRINTTAB(9); NF
·170 PRINT"[s -]  [s -]":PRINTTAB(9)"[s U
 ][c @][c @][s V]" PO
·171 PRINT"[HOME][19"[DOWN]"][BLUE]"TAB(3
 );"[4" "]" MH
·172 PRINT"[HOME][19"[DOWN]"][BLUE]"TAB(3
 );DP IP
·173 PRINT"[HOME][19"[DOWN]"][BLUE]"TAB(9
 );"[4" "]" BF
·174 PRINT"[HOME][19"[DOWN]"][BLUE]"TAB(9
 );DN-DP OO
·175 RETURN IM
·176 REM JD
·177 PRINT"[HOME][20"[DOWN]"][PURPLE]"TAB
 (3);"[s *][c T][c T][s R]":PRINTTAB(3); NM
·178 PRINT"[s -]  [s -]":PRINTTAB(3)"[s U
 ][c @][c @][s V]" ME
·179 RETURN IM
·180 REM JD
·181 IFLV=1THEN184 FG
·182 PRINT"[HOME][20"[DOWN]"][WHITE]"TAB(
 3);"TURN":PRINTTAB(3); GP
·183 PRINT"DECK":PRINTTAB(3)"OVER":GOTO18
 6 BL
·184 PRINT"[HOME][20"[DOWN]"][WHITE]"TAB(
 3);"E[3" "]":PRINTTAB(3); DG
·185 PRINT" N  ":PRINTTAB(3)"  D " HH
·186 RETURN IM
·187 REM JD
·188 PRINT"[HOME][23"[DOWN]"]"TAB(32)"CON
 CEDE" MO
·189 PRINT"[HOME]";:RETURN OF
·190 REM JD
·191 DATA234,173,000,220,141,144,003 AG
·192 DATA074,176,010,173,001,208,201 CD
·193 DATA052,144,003,206,001,208,173 OF
·194 DATA144,003,074,074,176,010,173 OC
·195 DATA001,208,201,228,176,003,238 FI
·196 DATA001,208,173,144,003,074,074 PE
·197 DATA074,176,034,173,016,208,074 DD
·198 DATA176,013,173,000,208,201,025 AH
·199 DATA144,021,206,000,208,024,144 GN
·200 DATA015,206,000,208,173,000,208 GD
·201 DATA201,255,208,005,169,000,141 BE
·202 DATA016,208,173,144,003,074,074 OI
·203 DATA074,074,176,029,173,016,208 DB
·204 DATA074,176,013,238,000,208,208 HG
·205 DATA018,169,001,141,016,208,024 CD
·206 DATA144,010,173,000,208,201,063 DL
·207 DATA176,003,238,000,208,173,144 PO
·208 DATA003,074,074,074,074,074,176 NE
·209 DATA001,096,160,243,162,161,232 FI
·210 DATA208,253,200,208,248,076,000 JP
·211 DATA193,000,000 KJ
```

```
•212 C2=.:FORI=49408TO49550:READB:C2=C2+B
 :POKEI,B:POKE1024,B:POKE16384,B:NEXT    LF
•213 IFC2<>16685THENPRINT"ERROR IN DATA L
 INES 11400-11500":STOP                  IH
•214 POKE49519,79:POKE49439,243          PH
•215 RETURN                              IM
•216 REM                                 JD
•217 DATA252,000,000,224,000,000,240,000 EN
•218 DATA000,152,000,000,140,000,000,012 LK
•219 DATA000,000,000,000,000,000,000,000 FG
•220 DATA000,000,000,000,000,000,000,000 FG
•221 DATA000,000,000,000,000,000,000,000 FG
•222 DATA000,000,000,000,000,000,000,000 FG
•223 DATA000,000,000,000,000,000,000,000 FG
•224 DATA000,000,000,000,000,000,000,000 FG
•225 FORI=22528TO22591:READB:POKEI,B:NEXT
 :RETURN                                 CJ
•226 REM                                 JD
•227 POKE53269,.:POKE53264,.:POKE53248,17
 0:POKE53249,150:POKE53287,6             AJ
•228 POKE17400,96:POKE53269,1:RETURN     BM
•229 REM READ JOYSTICK-SPRITE POSITION   DB
•230 SYS49408:YS=PEEK(53249):XS=PEEK(5324
 8)+256*(PEEK(53264)AND1)                CL
•231 XS=XS-24:YS=YS-51                   II
•233 CX=INT(XS/8):CY=INT(YS/8)           PC
•234 RETURN                              IM
•235 REM                                 JD
•236 JC=.                                GK
•237 IFCY>=22ANDCX>33THENJC=1            II
•238 IFCX>=3ANDCX<=7ANDCY>=19ANDCY<=22THE
 NJC=2                                   ML
•239 IFCX>=8ANDCX<=12ANDCY>=19ANDCY<=22TH
 ENJC=3                                  ML
•240 IFCX>=3ANDCX<=36ANDCY<=18ANDCY>=5THE
 NJC=4                                   JL
•241 IFCY<=4THENJC=5                     OO
•242 RETURN                              IM
•243 REM 1ST MOVE                        KJ
•244 AM=.                                HF
•245 GOSUB229:GOSUB235                   IJ
•246 ONJCGOTO248,249,254,256             NP
•247 GOSUB306:GOTO243                    HI
•248 GOTO45                              PP
•249 IFDP=.THENIFLV=.THENDP=DN-3:GOTO252 KP
•250 IFDP=.THEN374                       EO
•251 DP=DP-CG                            BO
•252 IFDP<.THENDP=.                      BN
•253 GOSUB155:GOTO243                    JA
•254 IFDP=DNTHEN247                      JL
•255 ZC=1:GOSUB155:AM=1:RETURN           DO
•256 REM                                 JD
•257 PN=7:IFCX<33THENPN=6                GA
•258 IFCX<28THENPN=5                     II
•259 IFCX<23THENPN=4                     HO
•260 IFCX<18THENPN=3                     IJ
•261 IFCX<O3THENPN=Z2                    BB
•262 IFCX<8THENPN=1                      EP
•263 P1=PN:IFNP(PN)=.THENGOSUB306:GOTO243 EK
•264 DC=.:FORI=1TONP(PN):IFP(PN,I)>52THEN
 DC=DC+Z1                                JO
•265 NEXT:D1=NP(PN)-DC+1                 KI
•266 CA=CY-5:CA=CA+D1:IFCA>NP(PN)THENCA=N
 P(PN)                                   MA
•267 SX=CA                               OF
•268 ZC=1:GOSUB111:RETURN                IH
•269 REM 2ND MOVE                        JJ
•270 GOSUB229:GOSUB235                   IJ
•271 ONJCGOTO273,274,275,277,308         LM
•272 GOSUB306:GOTO269                    JA
•273 GOTO248                             CK
•274 GOTO272                             CL
•275 IFAM=1THENZC=.:AM=.:BU=1:GOSUB155:RE
 TURN                                    AJ
•276 GOTO272                             CL
•277 REM DROP CARDS TO A PILE            EP
•278 P2=7:IFCX<33THENP2=6                OI
•279 IFCX<28THENP2=5                     HE
•280 IFCX<23THENP2=4                     GK
•281 IFCX<18THENP2=3                     GF
•282 IFCX<O3THENP2=Z2                    NF
•283 IFCX<8THENP2=1                      CD
•284 W2=P(P2,NP(P2))                     DG
•285 W2=W2-S4:S2=Z4:IFW2<O4THENS2=Z1     AD
•286 IFW2>=O4ANDW2<=T6THENS2=Z2          PC
•287 IFW2>T6ANDW2<=H9THENS2=Z3           AH
•288 IFW2>O3THENW2=W2-O3:GOTO288         IB
•289 IFAM=1THEN342                       DK
•290 W1=P(P1,SX)                         DI
•291 W1=W1-S4:GOSUB386                   GM
•292 IFNP(P2)=.ANDW1=O3THEN300           KG
•293 IFP1=P2THENBU=1:PN=P1:GOSUB111:RETUR
 N                                       IN
•294 IFW2<>W1+1THENGOSUB306:GOTO269      OD
•295 O1=1:IFS1=2ORS1=4THENO1=2           LM
•296 O2=1:IFS2=2ORS2=4THENO2=2           MI
•297 IFO1=1ANDO2=2THEN300                DK
•298 IFO1=2ANDO2=1THEN300                DE
•299 GOSUB306:GOTO269                    JA
•300 FORI=SXTONP(P1):NP(P2)=NP(P2)+1:P(P2
 ,NP(P2))=P(P1,I):NEXTI                  IL
•301 NP(P2)=NP(P2)+CQ                    OH
•302 NP(P1)=SX-1:P(P1,NP(P1))=P(P1,NP(P1)
 )ORS4                                   OP
•303 PN=P1:GOSUB111                      EL
•304 PN=P2:GOSUB111                      DI
•305 RETURN                              IM
•306 FORA=1TO20:POKE54296,15:FORT=1TO3:NE
 XT:POKE54296,0:FORT=1TO3:NEXT:NEXT      JD
•307 RETURN                              IM
•308 REM                                 JD
•309 IFAM=1THEN314                       DL
•310 IFSX<>NP(P1)THENGOSUB306:GOTO269    CJ
•311 W1=P(P1,SX)                         DI
•312 W1=W1-S4:GOSUB386                   GM
•313 ONS1GOTO326,330,334,338             LE
```

IMPORTANT! Letters on white background are **Bug Repellent** line codes. **Do not enter them!** Pages 117 and 118 explain these codes and provide other essential information on entering *Ahoy!* programs. Refer to these pages **before** entering any programs!

```
·314 REM                                    JD
·315 W1=DK%(DP+1)                           MF
·316 GOSUB386                               DB
·317 ONS1GOTO318,320,322,324                KM
·318 W2=H(NH):IFW2+1<>W1THENGOSUB306:GOTO
 269                                        JN
·319 NH=NH+Z1:H(NH)=W1:GOSUB365:GOSUB137:
 GOSUB155:RETURN                            CN
·320 W2=C(NC):IFW2+1<>W1THENGOSUB306:GOTO
 269                                        BB
·321 NC=NC+Z1:C(NC)=W1:GOSUB365:GOSUB137:
 GOSUB155:RETURN                            NJ
·322 W2=D(ND):IFW2+1<>W1THENGOSUB306:GOTO
 269                                        AN
·323 ND=ND+Z1:D(ND)=W1:GOSUB365:GOSUB137:
 GOSUB155:RETURN                            DN
·324 W2=S(NS):IFW2+1<>W1THENGOSUB306:GOTO
 269                                        DB
·325 NS=NS+Z1:S(NS)=W1:GOSUB365:GOSUB137:
 GOSUB155:RETURN                            BJ
·326 W2=H(NH):IFW2+1<>W1THENGOSUB306:GOTO
 269                                        JN
·327 NH=NH+1:H(NH)=W1                       JB
·328 NP(P1)=NP(P1)-Z1:P(P1,NP(P1))=P(P1,N
 P(P1))ORS4:GOSUB111:GOSUB137               EE
·329 RETURN                                 IM
·330 W2=C(NC):IFW2+1<>W1THENGOSUB306:GOTO
 269                                        BB
·331 NC=NC+1:C(NC)=W1                       BD
·332 NP(P1)=NP(P1)-Z1:P(P1,NP(P1))=P(P1,N
 P(P1))ORS4:GOSUB111:GOSUB137               EE
·333 RETURN                                 IM
·334 W2=D(ND):IFW2+1<>W1THENGOSUB306:GOTO
 269                                        AN
·335 ND=ND+1:D(ND)=W1                       FJ
·336 NP(P1)=NP(P1)-Z1:P(P1,NP(P1))=P(P1,N
 P(P1))ORS4:GOSUB111:GOSUB137               EE
·337 RETURN                                 IM
·338 W2=S(NS):IFW2+1<>W1THENGOSUB306:GOTO
 269                                        DB
·339 NS=NS+1:S(NS)=W1                       FD
·340 NP(P1)=NP(P1)-Z1:P(P1,NP(P1))=P(P1,N
 P(P1))ORS4:GOSUB111:GOSUB137               EE
·341 RETURN                                 IM
·342 REM                                    JD
·343 P2=7:IFCX<33THENP2=6                   OI
·344 IFCX<28THENP2=5                        HE
·345 IFCX<23THENP2=4                        GK
·346 IFCX<18THENP2=3                        GF
·347 IFCX<O3THENP2=Z2                       NF
·348 IFCX<8THENP2=1                         CD
·349 W2=P(P2,NP(P2))                        DG
·350 W2=W2-S4:S2=Z4:IFW2<O4THENS2=Z1        AD
·351 IFW2>=O4ANDW2<=T6THENS2=Z2             PC
·352 IFW2>T6ANDW2<=H9THENS2=Z3              AH
·353 IFW2>O3THENW2=W2-O3:GOTO353            JA
·354 W1=DK%(DP+1)                           MF
·355 GOSUB386                               DB
·356 IFW1=O3ANDNP(P2)=.THEN363              IH
·357 IFW2<>W1+1THENGOSUB306:GOTO269         OD
·358 O1=1:IFS1=2ORS1=4THENO1=2              LM
·359 O2=1:IFS2=2ORS2=4THENO2=2              MI
·360 IFO1=1ANDO2=2THEN363                   DF
·361 IFO1=2ANDO2=1THEN363                   DH
·362 GOSUB306:GOTO269                       JA
·363 NP(P2)=NP(P2)+Z1:P(P2,NP(P2))=DK%(DP
 +1)ORS4:GOSUB365                           KA
·364 GOSUB155:PN=P2:GOSUB111:RETURN         HM
·365 REM                                    JD
·366 FORA=DP+1TODN:DK%(A)=DK%(A+1):NEXT:D
 N=DN-1:RETURN                              GE
·367 END                                    IC
·368 REM LAS VEGAS OR REGULAR               IE
·369 PRINT"[6"[DOWN]"]":PRINT"LAS VEGAS S
 TYLE OR REGULAR??"                         DO
·370 PRINT"[DOWN][DOWN]PUSH UP FOR 'VEGAS
 ' AND DOWN FOR REGULAR"                    LI
·371 X=PEEK(56320):U=XAND1:V=XAND2:IFU=1A
 NDV=2THEN371                               EE
·372 IFU=.THENLV=1:CG=1:RETURN              II
·373 LV=.:CG=3:RETURN                       PG
·374 PRINT"[CLEAR]HAD YOU BOUGHT THE DECK
  FOR $50,"                                 HO
·375 PRINT"[DOWN]AT $5 FOR EVERY CARD YOU
  PUT AT THE TOP,"                          BK
·376 PRINT"[DOWN]YOU WOULD RECEIVE  $"(NH
 +NC+ND+NS)*5"BACK."                        NB
·377 PRINT"[DOWN]THIS LEAVES YOU WITH A N
 ET ";                                      MC
·378 IF((NH+NC+ND+NS)*5)-50>=0THENPRINT"P
 ROFIT":GOTO380                             DM
·379 PRINT"LOSS"                            DP
·380 PRINT"OF[3" "]$"ABS(((NH+NC+ND+NS)*5
 )-50)                                      OI
·381 GOTO45                                 PP
·382 X$="LAS VEGAS":PRINT:PRINT:PRINT:PRI
 NT"[WHITE]"                                KP
·383 FORI=1TO9:PRINT"[RIGHT]"MID$(X$,I,1)
 TAB(38)MID$(X$,I,1)                        NO
·384 NEXT                                   IA
·385 RETURN                                 IM
·386 S1=Z4:IFW1<O4THENS1=Z1                 JN
·387 IFW1>=O4ANDW1<=T6THENS1=Z2             JJ
·388 IFW1>T6ANDW1<=H9THENS1=Z3              BG
·389 IFW1>O3THENW1=W1-O3:GOTO389            EK
·390 RETURN                                 IM
```

MYSTERY AT MYCROFT MEWS

FROM PAGE 29

```
·10 REM "MYSTERY AT MYCROFT MEWS"           II
·20 S=54272:FORJ=0TO24:POKES+J,0:NEXT:POK
```

```
ES+24,15                                 NI
•30 POKE53280,0:POKE53281,0:POKE53283,6:P
RINT"[CLEAR]"CHR$(8)CHR$(142);          BF
•40 M$="[RVSON][c *][RIGHT][RIGHT][sEP][4
"[LEFT]"][DOWN] [c *][sEP] [4"[LEFT]"][D
OWN][4" "][4"[LEFT]"][DOWN] [s T][s Y] [
4"[LEFT]"][DOWN][RVSOFF][RED][c *][RVSON
][s M] [s M] [UP][LEFT] [LEFT][UP] [LEFT
][UP] [LEFT][UP][c *][RVSOFF][DOWN][DOWN
][WHITE]"                                NO
•50 PRINT"[CLEAR][DOWN][DOWN][c 3]"TAB(8)
M$"YSTERY AT":PRINT"[3"[DOWN]"][c 1]"TAB
(16)M$"YCROFT"                           PA
•60 PRINT"[3"[DOWN]"][YELLOW]"TAB(24)M$"E
WS":GOSUB20000                           KI
•100 FOR J=49152TO49220:READX:A=A+X:POKEJ
,X:NEXT:POKE785,0:POKE786,192            HG
•110 IFA<>8919THENPRINT"[CLEAR]ML DATA ER
ROR":STOP                                BM
•120 DEFFNR(X)=INT(RND(1)*X)+1:Y=RND(-TI)
:DEFFNSX(X)=-(X>6)                       DD
•140 V=53248:POKEV+28,1:POKEV+39,4:POKEV+
37,15:POKEV+38,6:POKEV+27,0:POKE2040,13  HJ
•150 FORJ=832TO894STEP3:POKEJ,0:POKEJ+1,0
:POKEJ+2,0:IFJ<=880THENREADA:POKEJ,A     GJ
•160 NEXT                                IA
•170 MX=FNR(3):MV=FNR(12):MW=FNR(12):MR=F
NR(12):IFMV=MRTHEN170                    OD
•180 I=12:DIML(74),U1(I),U2(I),G(I),AL(I)
,H(I),LO(I),SX(1,4)                      OE
•190 DIMAL$(5),IT$(16),B$(16),S$(I),H$(I)
:H$(0)="STAMP COLLECTOR"                 ON
•200 FORJ=1TO16:READB$(J),IT$(J)         ED
•210 I=FNR(16):IFL(58+I)>0THEN210        DA
•220 L(58+I)=J:L(12+J)=16:IFJ>12THEN250  BB
•230 READS$(J),H$(J):L(J)=J:AL(J)=FNR(6)-
1:H(J)=FNR(12):LO(J)=FNR(15)             NG
•240 L(J+12)=-LO(J):G(J)=FNR(7)-16*(RND(1
)<.5)-32*(RND(1)<.5)-64*(RND(1)<.5)      NF
•250 NEXT:L(30)=MX:L(49)=3ANDMW:L(50)=FNR
(40)+20:FORJ=0TO1:FORI=0TO3              PB
•260 READSX$(J,I):NEXTI,J:FORJ=0TO5:READA
L$(J):NEXT:FORJ=1TOMX:READIT$(13):NEXT   OK
•270 L(MV)=0:L(25)=-MV:AL(MR)=FNR(3)-1   BL
•275 LO(MW)=MR:IF-L(12+MW)=MRTHENL(12+MW)
=-FNR(15):GOTO275                        FM
•280 G(MV)=FNR(7)-8*(MX=2):IFMX<3THENG(MV
)=G(MV)OR6                               FN
•290 G(MR)=G(MR)OR2[UPARROW](MX+3):IFMX<3
THENG(MR)=G(MR)AND251                    LL
•300 PRINTTAB(8)"[3"[DOWN]"][PURPLE]HOW M
ANY PLAYERS? [WHITE](1-6)":POKE198,0     PC
•310 WAIT198,1:GETA$:A=VAL(A$):IFA<1ORA>6
THEN310                                  FL
•320 L(57)=A:FORJ=1TOA:L(36+J)=5:L(42+J)=
7:L(30+J)=17:NEXT:P=1:DIMSC$(A)          HA
•400 POKEV+17,27:PRINT"[CLEAR][DOWN][DOWN
]":POKEV+21,0                            OH
•410 FORY=1TO4:PRINT"[DOWN][3" "]";:FORX=
1TO4:H=(X*4-Y-6-2*(H=12))AND15           HG
•420 PRINT"[c 8][RVSON][sEP]  [RVSOFF][sE
P][s M][LEFT][LEFT][DOWN][c H][c N][LEFT
][LEFT][DOWN][s L][s @][5"[LEFT]"][UP]";
:POKE646,H:PRINT"[RVSON][3"[c T]"][3"[LE
FT]"][DOWN] [RIGHT] [RVSOFF][RIGHT][RIGH
T][UP][UP][3" "]";                       JL
•430 NEXT:PRINT"[3"[DOWN]"]":NEXT:PRINT"[
HOME][WHITE]PLAYER #"P"[DOWN]":IFL(57)-L
(58)=1THEN500                            KC
•440 PRINT"ENTER YOUR SECRET CODE!":POKE1
98,0:WAIT198,2:GETA$,B$                  CG
•450 IFSC$(P)=""THENSC$(P)=A$+B$         EK
•460 IFSC$(P)=A$+B$THEN500               EL
•470 J=USR(33000):GOSUB20000:GOSUB20100:G
OTO440                                   IK
•500 X=L(36+P):Y=L(42+P):POKES+1,33:POKES
+5,9:POKES+6,0                           NL
•510 POKE198,0:GOSUB20100:PRINT"MOVE TO B
UILDING DESIRED":GOSUB20500              DK
•520 GOSUB20300:GOSUB20100:IF(XAND1)OR(YA
ND1)THEN520                              FI
•530 J=2*Y+X/2-8:B=L(58+J):IFB<>MVTHEN600 IP
•540 IFL(26)=P+100ORABS(L(26))=MVORL(30+P
)=MVTHEN600                              MF
•550 PRINT"THAT BUILDING IS LOCKED. TRY A
NOTHER.":GOTO520                         EF
•600 L(30+P)=B:L(36+P)=X:L(42+P)=Y:POKES+
6,96:POKES+4,33:FORJ=0TO12:POKES+1,33-J  EI
•610 U2(J)=0:U1(J)=0:NEXT:POKES+4,0:POKEV
+21,0:PRINT"[CLEAR][c 7]YOU ARE AT THE "
B$(B)".[c 8]"                            FO
•620 PRINT"YOU HAVE ";:IFL(50+P)THENPRINT
"THE "IT$(L(50+P))".":GOTO640            OL
•630 PRINT"NO ITEM."                     PP
•640 I=0:PRINT"[DOWN]YOU SEE:[c 5]":FORJ=
1TO12                                    CE
•645 IFL(J)=BTHENPRINTS$(J):U2(1)=U2(1)+1
:I=1:U2(5)=J                             OD
•650 NEXT:FORJ=1TO16:IFL(J+12)=BTHENPRINT
"THE "IT$(J):U2(2)=U2(2)+1:I=1           HI
•660 NEXT:FORJ=1TOL(57):IFJ=PORL(J+30)<>B
THEN690                                  NH
•670 PRINT"PLAYER #"J;:I=1:U2(3)=U2(3)+1:
IFL(50+J)=0THENPRINT:GOTO690             BH
•680 PRINT"<WITH AN ITEM>":U2(4)=U2(4)+1 GF
•690 NEXT:IFI=0THENPRINT"NOTHING WORTH ME
NTIONING."                               KA
•695 PRINT"[DOWN][c 8]PRESS BUTTON FOR TU
RN OPTIONS":GOSUB20400                   OO
•700 PRINT"[CLEAR][c 3]TURN OPTIONS[3"."]
[4"[DOWN]"][WHITE]":MY=1:Y=1:POKEV+17,91
:U1(1)=1:PRINT"SEARCH"                   NM
•720 IFU2(1)THENPRINT"INTERROGATE SUSPECT
":MY=2:U1(2)=2                           IJ
```

```
•730 IFU2(2)ORU2(4)THENPRINT"TAKE ITEM":M
 Y=MY+1:U1(MY)=3                          JN
•740 IFB=16AND(U2(2)ORL(50+P))THENPRINT"F
 INGERPRINT ITEM":MY=MY+1:U1(MY)=4        II
•750 IFB=16THENPRINT"ACCUSE A SUSPECT":MY
 =MY+1:U1(MY)=5                           AL
•760 IFB=14THENPRINT"READ CORONER'S REPOR
 T":MY=MY+1:U1(MY)=6                      KD
•770 IFL(25)=100+PORL(25)=BTHENPRINT"READ
  "IT$(13):MY=MY+1:U1(MY)=7               PK
•780 IFL(50+P)THENPRINT"DROP OR HIDE ITEM
  CARRIED":MY=MY+1:U1(MY)=8               EK
•790 IFB<3ORB=7THENPRINT"EXAMINE "B$(B)"
 RECORDS":MY=MY+1:U1(MY)=9                NO
•800 GOSUB20100:PRINT"CHOOSE ONE":GOSUB20
 600:I=U1(Y):Y=1:PRINT"[CLEAR][c 7]";     BE
•810 ONIGOSUB1000,2000,1400,1100,5000,120
 0,1500,1300,1600:J=FRE(0)                EN
•900 POKE214,22:PRINT:PRINT"[YELLOW] [3"*
 "] PRESS BUTTON TO END YOUR TURN [3"*"]"
 ;:GOSUB20400                             FM
•910 PRINT"[CLEAR][8"[DOWN]"][c 7]"TAB(12
 )"PLAYER #"P:PRINTTAB(12)"[DOWN]END OF T
 URN"                                     DI
•915 J=USR(33000):GOSUB20000              LB
•920 P=P+1:IFP>L(57)THEN950               IL
•930 IFL(30+P)>0THEN400                   KD
•940 GOTO920                              CL
•950 FORJ=1TO12:L(J)=FNR(16):IFL(J)=MVTHE
 NL(J)=J                                  CJ
•960 NEXT:FORJ=31TO36:IFL(J)<13THENL(L(J)
 )=L(J)                                   CC
•970 NEXT:L(MV)=0:P=1:GOTO930             MO
•1000 PRINT"SEARCH[DOWN][DOWN]":PRINT"[c
 8]YOU FIND:[DOWN][c 5]":I=0              CG
•1010 FORJ=1TO16:IFL(J+12)<>-BTHEN1030    HK
•1020 IFRND(1)<.6ORL(50+P)=15THENI=1:PRIN
 T"THE "IT$(J):L(J+12)=B                  MI
•1030 NEXT:IFI=0THENPRINT"NOTHING OF INTE
 REST":RETURN                             FL
•1040 PRINT"[DOWN][c 7]TAKE AN ITEM?  [WH
 ITE]";:GOSUB20700:IFYN=0THENRETURN       OJ
•1050 U2(4)=0:PRINT"[CLEAR][c 7]";:Y=1:GO
 TO1400                                   MM
•1100 PRINT"FINGERPRINT ITEM":PRINT"[DOWN
 ][c 8]WHICH ONE?[WHITE][DOWN][DOWN]":MY=
 0                                        CF
•1110 FORJ=1TO16:IFL(J+12)=BTHENPRINT"THE
  "IT$(J):MY=MY+1:U1(MY)=J                OM
•1120 NEXT:I=L(P+50):IFITHENPRINT"THE "IT
 $(I):MY=MY+1:U1(MY)=I                    LL
•1130 GOSUB20600:I=U1(Y):PRINT"[CLEAR][c
 7]LAB REPORT:":PRINT"[DOWN][WHITE]THE "I
 T$(I)                                    EH
•1140 PRINT"BEARS THE FINGERPRINTS OF":L(
 I+12)=B:IFL(P+50)=ITHENL(P+50)=0         ON
•1150 IFI<13THENIFL0(I)<13THENPRINTS$(L0(
 I))".":RETURN                            LH
•1160 IFI=13THENPRINTS$(MV)".":RETURN     LL
•1170 PRINT"NO SUSPECT.":RETURN           MP
•1200 PRINT"CORONER'S REPORT:[WHITE][DOWN
 ][DOWN]":X=ABS(MV>6):PRINT"SUBJECT IS "S
 X$(X,0)","                               AH
•1210 PRINT"APPROXIMATELY"L(50)"YEARS OF
 AGE.":J=USR(34000+L(49)*10)              IF
•1220 READT$:PRINT"[DOWN]SUBJECT APPARENT
 LY WAS "T$                               EK
•1230 READT$:PRINT"BY A "T$".":RETURN     NJ
•1300 PRINT"DROP OR HIDE ITEM CARRIED":PR
 INT"[DOWN][DOWN][c 8]DO YOU WANT TO HIDE
  THE"                                    AJ
•1310 I=L(50+P):PRINTIT$(I)"?  [WHITE]";:
 GOSUB20700:L(50+P)=0                     CD
•1320 PRINT:PRINT"[DOWN][c 8]YOU DROP THE
  "IT$(I)"."                              IG
•1330 L(12+I)=B*(1+YN*2):IFYNTHENPRINT"IT
 EM HIDDEN."                              IK
•1340 FORI=1TO200:NEXT:POKE198,0:RETURN   GB
•1400 PRINT"TAKE ITEM":PRINT"[DOWN][c 8]W
 HICH ONE?[WHITE][DOWN][DOWN]":MY=0       KC
•1410 FORJ=1TO16:IFL(J+12)=BTHENPRINT"THE
  "IT$(J):MY=MY+1:U1(MY)=J                OM
•1420 NEXT:IFU2(4)=0THEN1450              LH
•1430 FORJ=1TOL(57):IFJ=PORL(J+30)<>BORL(
 J+50)=0THENNEXT:GOTO1450                 CF
•1440 PRINT"PLAYER #"J"[LEFT]'S ITEM":MY=
 MY+1:U1(MY)=100+J:NEXT                   HG
•1450 GOSUB20600:I=U1(Y):IFI>100THENI=L(I
 -50):L(U1(Y)-50)=0                       DF
•1460 PRINT"[CLEAR]":J=L(50+P):IFJTHENPRI
 NT"YOU DROP THE "IT$(J)".":L(J+12)=B     KJ
•1470 PRINT"YOU TAKE THE "IT$(I)".":L(50+
 P)=I:L(I+12)=100+P:RETURN                BI
•1500 PRINT"EXTRACT FROM THE "IT$(13):PRI
 NT"OF "S$(MV)":[WHITE][DOWN][DOWN]"      AE
•1510 J=USR(34000+L(30)*100)              OL
•1520 READT$:I=VAL(T$):IFI=0THENPRINTT$:G
 OTO1520                                  NF
•1530 PRINT"[c 5]":FORJ=1TO12:IFIANDG(J)T
 HENPRINT"[4" "]"S$(J)                    ED
•1540 NEXT:PRINT:J=FRE(0):RETURN          IF
•1600 PRINT"EXAMINE "B$(B)" RECORDS":PRIN
 T"[DOWN][c 8]ABOUT WHOM?[WHITE][DOWN][DO
 WN]"                                     JJ
•1610 FORJ=1TO12:PRINTS$(J):NEXT:MY=12:GO
 SUB20600                                 OC
•1620 PRINT"[CLEAR][c 7]"B$(B)" REPORT":P
 RINT"ON "S$(Y)":[WHITE][DOWN]":ONBGOTO16
 50,1690                                  II
•1630 J=USR(34400):FORJ=1TO(G(Y)AND7):REA
 DT$:NEXT                                 HG
•1640 PRINTTAB(8)T$" ACCOUNT":RETURN      FD
•1650 IF(G(Y)AND40)=0THENPRINT"SUBJECT NO
 T IN RECORDS.":RETURN                    NF
```

```
•1660 IFG(Y)AND8THENPRINT"SUBJECT RECENTL
Y DREW UP A NEW WILL.":RETURN CH
•1670 PRINT"SUBJECT CAN EXPECT TO INHERIT
":PRINT"SUBSTANTIAL WEALTH IN THE NEAR" CA
•1680 PRINT"FUTURE.":RETURN LJ
•1690 IF(G(Y)AND64)=0THENPRINT"SUBJECT NO
T IN FILES":RETURN MM
•1700 PRINT"PATIENT SEVERELY PARANOIAC, E
NVIOUS AND" ED
•1710 PRINT"RESENTFUL OF OTHERS. MAY SHOW
 SUDDEN" IC
•1720 PRINT"HOMICIDAL FURY.":RETURN LH
•2000 Y=1:Q=3:H=0:Q4=-1:IFU2(1)=1THENQ1=U
2(5):GOTO2100 BO
•2010 PRINT"INTERROGATE A SUSPECT":PRINT"
[DOWN][c 8]WHICH ONE?[DOWN][DOWN][WHITE]
":MY=0 PB
•2020 FORJ=1TO12:IFL(J)=BTHENPRINTS$(J):M
Y=MY+1:U1(MY)=J AA
•2030 NEXT:GOSUB20600:Q1=U1(Y):Y=1 ID
•2100 PRINT"[CLEAR][c 7]INTERROGATE "S$(Q
1):MY=13:POKE198,0:I=12:IFQ4=0THEN2120 JP
•2110 PRINT"[DOWN][c 8]ABOUT WHOM?[DOWN][
DOWN][WHITE]":FORJ=1TOI:PRINTS$(J):NEXT HN
•2115 PRINT"<AN ITEM>":GOTO2130 JJ
•2120 PRINT"[DOWN][c 8]ABOUT WHAT ITEM?[D
OWN][DOWN][WHITE]":FORJ=1TOI:PRINT"THE "
IT$(J):NEXT IA
•2125 PRINT"<A SUSPECT>" PO
•2130 IFH<>0THENPOKE782,Y:SYS49189:H=0 EF
•2140 GOSUB20600:H=1:IFY=13THENH=0:Q4=NOT
Q4:GOTO2100 IN
•2150 Q2=Y:Q=Q-1:YN=(RND(1)<.75):GOSUB201
00:ONQ4+2GOSUB2200,3000 KN
•2160 FORJ=1TO100:NEXT:POKE198,0:IFQTHEN2
130 DJ
•2170 J=FRE(0):RETURN AC
•2200 Q3=FNR(12):IFQ3=MVORQ3=Q1THEN2200 OB
•2210 T$=S$(Q2):IFRND(1)>.2ORL(28)=100+PT
HEN2240 EC
•2220 L(29)=(L(29)+1)AND1:IFL(29)THENPRIN
T"I NEVER GOSSIP.":RETURN OF
•2230 PRINT"I DON'T HAVE TIME TO TALK NOW
.":RETURN HH
•2240 X=FNSX(Q2):IFQ2=Q1THEN2500 PI
•2250 IFQ2=MVTHEN2400 NP
•2260 ONFNR(7)GOTO2290,2300,2300,2320,232
0,2360 LK
•2270 IFAL(Q1)<3ORAL(Q1)<>AL(Q2)THEN2285 PO
•2280 PRINT"I WAS WITH "T$:PRINT"ON THE N
IGHT OF THE MURDER.":RETURN OP
•2285 PRINT"I CAN'T CONFIRM "SX$(X,3)" AL
IBI.":RETURN BD
•2290 PRINTT$" DID IT, I'M SURE!":RETURN IL
•2300 PRINTT$:I=FNR(3):J=2*I-1:IFG(Q2)AND
2[UPARROW](3+I)THENJ=J-1 LI
•2310 GOSUB20800:RETURN FK
•2320 GOSUB20900:IFRND(1)>.5THEN2370 FE
•2340 PRINTT$" OWNS":A=-1*YN:IFA=0THENPRI
NT"A NEW CAR.":RETURN GD
•2350 PRINT"THE "IT$(A)".":RETURN MI
•2360 A=H(Q2) AA
•2370 PRINTT$" IS":PRINT"A "H$(A)".":RETU
RN PA
•2400 ONFNR(3)GOTO2440,2470 FJ
•2410 IFFNR(3)=1THENPRINT"HOW TRAGIC THAT
":PRINTSX$(X,1)"'S GONE.":RETURN CG
•2420 IFFNR(2)=1THENPRINT"I THINK IT WAS
SUICIDE.":RETURN PN
•2430 PRINT"I THINK THAT "S$(Q3):PRINT"MU
RDERED "SX$(X,2)".":RETURN DB
•2440 I=FNR(4)-1:IFYNTHENI=L(49) FB
•2450 PRINT"I HEARD THAT "SX$(X,1)" WAS K
ILLED":J=USR(34000+I*10) EI
•2460 READT$,T$:PRINT"BY A "T$".":RETURN GC
•2470 I=FNR(3):IFYNTHENI=L(30) JB
•2480 PRINT"I HEARD THAT "SX$(X,1)" WAS":
J=I+5:GOSUB20800:RETURN EE
•2500 ONFNR(4)GOTO2530,2540,2550 JL
•2510 GOSUB20900:IFA=0THENPRINT"I OWN A N
EW CAR.":RETURN FM
•2520 PRINT"I OWN THE":PRINTIT$(A)".":RET
URN FJ
•2530 PRINT"I WAS "AL$(AL(Q1)):PRINT"ON T
HE NIGHT OF THE CRIME.":RETURN NE
•2540 A=H(Q1):GOTO2560 KC
•2550 GOSUB20900 IM
•2560 PRINT"I AM A "H$(A)".":RETURN KN
•3000 ONFNR(5)GOTO3030,3060,4030,4050 MK
•3010 PRINT"I ";:IFNOT(LO(Q2)=Q1)OR(Q1=MR
)THENPRINT"DO NOT "; JG
•3020 PRINT"OWN THE":PRINTIT$(Q2)".":RETU
RN EM
•3030 IFYN=0OR(Q2=MWANDQ1=MR)ORLO(Q2)>12T
HEN3050 JN
•3035 IFLO(Q2)=Q1THEN3010 JB
•3040 PRINTS$(LO(Q2))" OWNS":PRINT"THE "I
T$(Q2)".":RETURN ED
•3050 PRINT"I'M NOT SURE WHO OWNS":PRINT"
THE "IT$(Q2)".":RETURN MC
•3060 I=FNR(57):X=ABS(L(12+Q2)):IFX<100TH
EN4000 DH
•3070 X=X-100:IFYN=0THENX=I NM
•3080 IFX=PTHENPRINT"YOU HAVE THE "IT$(Q2
)",":PRINT"DON'T YOU?":RETURN GC
•3090 PRINT"I THINK THAT PLAYER #"X"HAS":
PRINT"THE "IT$(Q2)".":RETURN JH
•4000 IFYN=0OR(Q2=MWANDQ1=MR)THENJ=9:GOTO
20800 PF
•4010 IFX=Q1THENPRINT"I'VE GOT THE":PRINT
IT$(Q2)".":RETURN AK
•4020 PRINT"I SAW THAT AT THE ":PRINTB$(X
)".":RETURN CN
•4030 PRINT"THE "IT$(Q2)" WAS ";:I=(Q2=MW
)=YN:IFI=0ORQ1=MRTHENPRINT"NOT"; KH
•4040 PRINT:PRINT"THE MURDER WEAPON.":RET
```

```
URN                                              NK
•4050 X=0:FORJ=1TO12:IFH(J)=Q2ANDRND(1)>.
 5THENX=J                                        EI
•4060 NEXT:I=L0(Q2):IFI>12THENI=0                MI
•4070 IFRND(1)<.5THENX=I                         NB
•4080 IFX=0THENJ=9:GOTO20800                     HP
•4090 PRINTS$(X)" IS":PRINT"A "H$(Q2)".":
 RETURN                                          LK
•5000 PRINT"[c 3]ACCUSE!":PRINT"[DOWN][c
 8]WHO WAS KILLED?[WHITE][DOWN][DOWN]":MY
 =12:Q=-1                                        AK
•5010 FORJ=1TO12:PRINTS$(J):NEXT:Y=1:GOSU
 B20600:Q=Q*(Y=MV)                               II
•5020 PRINT"[CLEAR][YELLOW]ACCUSE!":PRINT
 "[DOWN][c 8]WHO KILLED "S$(Y)"?[DOWN][DO
 WN][WHITE]":Y=1                                 FK
•5030 FORJ=1TO12:PRINTS$(J):NEXT:GOSUB206
 00:H=Y:Q=Q*(H=MR)                               AA
•5040 PRINT"[CLEAR][c 6]ACCUSE!":PRINT"[D
 OWN][c 8]WHAT WEAPON DID":PRINTS$(H)" US
 E?[DOWN][WHITE]":Y=1                            DA
•5050 FORJ=1TO12:PRINT"THE "IT$(J):NEXT:G
 OSUB20600:Q=Q*(Y=MW)                            DN
•5060 PRINT"[CLEAR][c 7]ACCUSE!":PRINT"[D
 OWN][c 8]WHY DID "S$(H):PRINT"COMMIT MUR
 DER?[DOWN][WHITE]":Y=1                          BF
•5070 X=FNSX(H):FORA=0TO4STEP2:J=A:PRINTS
 X$(X,1)" ";:GOSUB20800:NEXT                     IB
•5080 MY=3:Y=1:GOSUB20600:Q=Q*(Y=L(30))          FE
•5100 PRINT"[CLEAR]THE POLICE ARE ARRESTI
 NG":PRINTS$(H)"[5"."]":POKES+5,13:POKES+
 6,0                                             IN
•5110 POKES+4,65:FORJ=1TO20:POKES+1,16+16
 *(JAND1):FORI=1TO250:NEXT:NEXT                  EB
•5120 POKES+4,0:POKEV+17,27:FORJ=1TO600:N
 EXT:IFQTHEN5200                                 LC
•5130 PRINT"[DOWN]NO[3"."]THAT WAS A FALS
 E ARREST.":PRINT"YOU'RE OUT OF THE GAME!
 "                                               PE
•5140 L(30+P)=0:IFL(50+P)THENL(12+L(50+P)
 )=B:L(50+P)=0                                   IF
•5150 J=USR(33200):GOSUB20000:L(58)=L(58)
 +1:IFL(58)<L(57)THENRETURN                      IE
•5160 RESTORE:PRINT"[DOWN][c 3]ALL PLAYER
 S HAVE BEEN ELIMINATED!"                        FP
•5170 PRINT"[WHITE]NOBODY WINS!":PRINT"[P
 URPLE][6"[c T]"]":GOTO5300                      MG
•5200 PRINT"[DOWN]YES[3"."]YOU'VE CRACKED
  THE CASE!":PRINT"PLAYER #"P"WINS THE GA
 ME!"                                            BL
•5210 J=USR(33100)                               PH
•5300 GOSUB20000:PRINT"[DOWN][DOWN][WHITE
 ]HERE IS THE SOLUTION:[DOWN][c 5]":PRINT
 S$(MR)" MURDERED"                               MF
•5310 PRINTS$(MV)" USING THE":PRINTIT$(MW
 )", BECAUSE "SX$(FNSX(MR),1)                    EF
•5320 J=L(30)*2-2:GOSUB20800:FORJ=1TO4E3:
 NEXT:END                                        HH
```

```
•20000 POKES+5,9:POKES+6,192:POKES+3,8:T=
 TI                                              LN
•20010 IFT>TITHEN20010                           NM
•20020 POKES+4,64:READJ:IFJ=0THEN20040           AF
•20030 READX,Y:POKES+1,X:POKES,Y:T=T+J*9:
 POKES+4,65:GOTO20010                            KJ
•20040 POKES+4,0:RETURN                          PH
•20100 SYS49178:PRINT"[HOME][DOWN][DOWN][
 c 8]";:RETURN                                   LM
•20200 J=NOT(PEEK(56320)ANDPEEK(56321))AN
 D31:A=0:IFPEEK(198)THENGETA$:A=ASC(A$)          FD
•20210 BT=(JAND16)OR(A=13):IFBTTHENRETURN        CH
•20220 DX=SGN((JAND8)-(A=135)-(JAND4)+(A=
 134))                                           PK
•20230 DY=SGN((JAND2)-(A=136)-(JAND1)+(A=
 133)):IFDXORDYTHENRETURN                        OL
•20240 GOTO20200                                 IF
•20300 GOSUB20200:IFBTTHENRETURN                 DG
•20310 IFABS(DX+X-5)>3THENDX=0                   AF
•20320 IFABS(DY+Y-7)>3THENDY=0                   GI
•20330 IFDX=0ANDDY=0THEN20300                    MI
•20340 FORI=1TO4:X=X+DX/4:Y=Y+DY/4:GOSUB2
 0500:POKES+1,30+3*(1ANDI)                       MA
•20350 POKES+4,17:FORJ=1TO40:NEXT:POKES+4
 ,0:NEXT:GOTO20300                               BK
•20400 POKE198,0                                 KB
•20410 GOSUB20200:IFBT=0THEN20410                KI
•20420 RETURN                                    IM
•20500 POKEV,X*32-1:POKEV+1,Y*20+12:POKEV
 +21,1:RETURN                                    FE
•20600 POKE782,Y:SYS49189                        PK
•20610 GOSUB20200:IFBTTHENRETURN                 DG
•20620 X=Y+DY:IFX<1ORX>MYTHENX=-(X>MY)-MY
 *(X<1)                                          KF
•20630 IFX=YTHEN20610                            JC
•20640 POKE782,Y:SYS49189:Y=X:GOTO20600          CF
•20700 POKE190,0:YN=-1                           PH
•20710 PRINTCHR$(-18*YN)"YES[RVSOFF] "CHR
 $(YN*2+18)"NO [RVSOFF][7"[LEFT]"]";             EC
•20720 GOSUB20200:IFBTTHENRETURN                 DG
•20730 IFDXTHENYN=NOTYN:GOTO20710                GB
•20800 J=USR(35000+10*J):READT$:PRINTT$".
 ":RETURN                                        MC
•20840 GOTO20720                                 IA
•20900 A=0:X=1:FORJ=1TO12:IFL0(J)=Q2ANDRN
 D(1)<1/XTHENA=J:X=X+1                           KB
•20910 NEXT:RETURN                               EJ
•30000 DATA8,11,48,12,10,143,4,9,104,4,10
 ,143,4,11,48,12,13,78                           KL
•30010 DATA4,14,24,12,8,97,0                     BB
•31000 DATA32,247,183,32,19,166,144,13,16
 5,95,233,1                                      AH
•31010 DATA133,65,165,96,233,0,133,66,96,
 162,17,76                                       AN
•31020 DATA55,164,169,32,160,80,136,153,8
 0,4,208,250                                     GO
•31030 DATA96,169,4,133,252,169,160,24,10
 5,40,144,2                                      MO
```

```
•31040 DATA230,252,136,208,246,133,251,16
0,40,136,177,251                          ON
•31050 DATA73,128,145,251,192,0,208,245,9
6                                         PB
•32000 DATA20,20,20,0,170,170,170,170,190
,190,190,125,60,60,60,60,85               HO
•32100 DATALAW OFFICE,ELEPHANT GUN,PHYLLI
S WRIT,BIG-GAME HUNTER                    GO
•32110 DATAMENTAL HEALTH CLINIC,ANTIQUE C
ROSSBOW,DR. WILMA LIBIDO                  AG
•32120 DATA"[LEFT]N ARCHERY EXPERT",DELIC
ATESSEN,BASEBALL BAT,HANNAH NOSH,SPORTS
FAN                                       JH
•32130 DATATHEATRE,BOX OF RAT POISON,EDIT
H SCRIM,PEST EXTERMINATOR,HOTEL           NH
•32140 DATAARQUEBUS,CHERYL VALISE,GUN COL
LECTOR,ART GALLERY,SCIMITAR               AP
•32150 DATADESIREE OCHRE,MASTER FENCER,BA
NK,BRASS CANDLESTICK,EMILIO LUCRE         NH
•32160 DATACANDELABRA FANCIER,LIBRARY,JAR
 OF SLEEPING PILLS,MORTON SCREED          KJ
•32170 DATA"[LEFT]N INSOMNIAC",BAKERY,.45
 AUTOMATIC,JOSEPH CRULLER,MARKSMAN,PAWN
SHOP                                      KP
•32180 DATAKNITTING NEEDLE,NORMAN HOCK,CH
AMPION KNITTER,CHURCH,BALL-PEEN HAMMER    JJ
•32190 DATAREV. ARTHUR TRANSEPT,TOOL COLL
ECTOR,PUB,VIAL OF SNAKE VENOM             FF
•32200 DATAEDGAR QUAFF,HERPETOLOGIST,SHOP
PING MALL,ITEM,MORGUE,MASTER KEY          AO
•32210 DATAWAREHOUSE,MAGNIFYING LENS,POLI
CE STATION,POLICE BADGE                   IF
•32300 DATAFEMALE,SHE,HER,HER,MALE,HE,HIM
,HIS                                      KG
•32400 DATAAT HOME ALONE,SICK IN BED,OUT
OF TOWN,AT A PARTY                        HL
•32410 DATAAT THE THEATRE,IN CHURCH       GA
•32900 DATALEDGER,WILL,DIARY              EB
•33000 DATA2,15,210,2,14,24,2,12,143,2,11
,218,4,9,247,4,7,233,4,10,143,0           BP
•33100 DATA4,28,49,4,37,162,4,42,62,2,47,
107,4,37,162,2,42,62,2,47,107,2,37,162    GP
•33110 DATA2,42,62,4,56,99,2,42,62,2,47,2
07,1,42,62,1,37,162,2,42,62,8,28,49,0     FD
•33200 DATA2,25,30,2,23,181,2,31,165,2,35
,134,2,47,107,2,37,162,2,35,134           HI
•33210 DATA2,47,107,2,37,162,2,35,134,2,3
1,165,2,29,223,4,31,165,0                 CC
•34000 DATAPOISONED,"[LEFT]N UNIDENTIFIED
 TOXIN"                                   NP
•34010 DATASHOT,GUN                       OD
•34020 DATAIMPALED,SHARP WEAPON           AK
•34030 DATABLUDGEONED,BLUNT INSTRUMENT    PN
•34100 DATAI HAVE THREATENED TO TELL THE
POLICE                                    HC
•34110 DATAALL THAT I KNOW ABOUT THEIR CR
IMINAL                                    AG
•34120 DATAACTIVITIES IF ANY OF THESE PEO
PLE                                       GO
•34130 DATADOESN'T PAY ME ANOTHER $9000 B
Y,MIDNIGHT SUNDAY...,16                   OO
•34200 DATAI HEREBY BEQUEATH ONE HUNDRED
MILLION                                   DM
•34210 DATADOLLARS TO EACH OF THESE LOYAL
AND                                       FI
•34220 DATAFAITHFUL FRIENDS...,32         IC
•34300 DATADEAR DIARY---,I FOUND ANOTHER
ANONYMOUS LETTER IN THE                   FJ
•34310 DATAMAIL THIS MORNING. MY LIFE IS
IN DANGER                                 KO
•34320 DATAFROM A JEALOUS LUNATIC. THE HA
NDWRITING                                 NC
•34330 DATACOULD BE THAT OF ANY OF THESE
PEOPLE...,64                              CH
•34400 DATANO,OVERDRAWN,MINIMUM,AVERAGE,S
IGNIFICANT,SUBSTANTIAL,MAXIMUM            CE
•35000 DATAWAS BEING BLACKMAILED          KD
•35010 DATAHAS NO SECRETS TO HIDE         BF
•35020 DATAWILL INHERIT MILLIONS          KE
•35030 DATAIS NOT A LEGATEE               CM
•35040 DATAIS A HOMICIDAL MANIAC          KI
•35050 DATAIS AS SANE AS I AM             DP
•35060 DATAA BLACKMAILER                  OP
•35070 DATAEXTREMELY WEALTHY              HA
•35080 DATARECEIVING DEATH THREATS        KC
•35090 DATAI DON'T KNOW WHERE THAT OBJECT
IS                                        FF
```

SHOTGUN

FROM PAGE 97 SHOTGUN LOADER

```
•2 REM        "SHOTGUN LOADER"            MF
•4 REM       BY R. HAROLD DROID           MH
•10 IFSG=1THEN30                          BL
•20 SG=1:LOAD"SHOTGUN SHELL",8,1          DO
•30 PRINTCHR$(13)"THIS PROGRAM CREATES AN
 AUTO-MENU"                               LO
•33 PRINT"  FOR A GIVEN DISK"CHR$(13)     JG
•34 PRINT"SYS51200 ENABLES YOUR NEW PROGR
AM:"                                      AH
•35 PRINT" SHIFT/RESTORE BREAKS TO MENU"  DF
•36 PRINT" SPACE/RESTORE RERUNS WITHOUT R
ELOADING"                                 LG
•37 PRINT" RUN-STOP/RESTORE DISABLES"     DH
•38 PRINT" COMMODORE/RESTORE DOES NON-DIS
ABLING"                                   EF
•39 PRINT"[3" "]WARM START"CHR$(13)       KA
•40 PRINT"INSERT A DISK CONTAINING"       EC
•50 PRINT"  BASIC PROGRAM FILES"          JL
•60 PRINT"  THEN PRESS A KEY"CHR$(13)     PC
•70 POKE198,0:WAIT198,1                   IE
•80 REM ******* READ DIRECTORY ********   HA
•85 PN=1:PT=51769                         EF
•90 OPEN1,8,0,"$0"                        GF
•100 FLAG=1:REM FLAG SET FOR READING DISK
 NAME                                     BD
```

```
•110 F$="":GET#1,B$,B$,B$,B$                PF
•120 GET#1,B$:IF ST<>0 THEN320              LN
•130 IF B$<>CHR$(34) THEN120                PA
•140 GET#1,B$:IF B$<>CHR$(34)THEN F$=F$+B
 $:GOTO140                                  FL
•150 GET#1,B$:IF B$=CHR$(32) THEN150        DJ
•160 GET#1,B$:IFB$<>""THEN160               LE
•170 IF ST<>0 THEN320                       AG
•180 IFFLAG=0THEN230                        NA
•190 FLAG=0:DN$(DN)=F$:PRINT"DISK NAME: "
 DN$(DN)CHR$(13)                            JO
•200 PRINT"INDICATE WHETHER A FILE SHOULD
 "                                          OL
•210 PRINT"  BE INCLUDED IN THE AUTO-MENU
 "                                          PJ
•215 PRINT"  BY PRESSING Y OR N"            PE
•218 PRINT"  CAPACITY IS 26 FILES!"CHR$(1
 3)                                         NM
•220 GOTO110                                CC
•230 PRINTF$;                               ED
•240 POKE198,0:WAIT198,1:GETA$:IFA$="Y"TH
 EN270                                      OB
•250 IFA$="N"THENPOKE211,0:PRINT"[16" "]"
 ;:POKE211,0:GOTO110                        PL
•260 GOTO240                                CC
•270 PRINT                                  JJ
•275 IFLEN(F$)<>16THENF$=F$+CHR$(160):GOT
 O275                                       OM
•280 FORA=1TO16:POKEPT,ASC(MID$(F$,A,1)):
 PT=PT+1:NEXT                               HK
•290 PN=PN+1:IFPN>27THEN400                 IP
•310 GOTO110                                CC
•320 CLOSE1:POKE51768,PN:PT=PT+1            IB
•322 PRINT:INPUT"AUTO MENU FILENAME  SHOT
 GUN[9"[LEFT]"]";PG$                        BO
•330 OPEN8,8,8,PG$+",P,W"                   GL
•340 PRINT#8,CHR$(0);:PRINT#8,CHR$(192);    LP
•350 FORA=51200TOPT:PRINT#8,CHR$(PEEK(A))
 ;:NEXT                                     CH
•360 CLOSE8:END                             JH
•400 PRINT"MENU CAPACITY EXCEEDED!  TRY A
 GAIN[3"."]":CLOSE1:GOTO85                  GL
```

SHOTGUN SHELL*

Beginning address in hex: C800 Ending address in hex: CA36

```
C800: A9 00 8D 37 CA 20 EA C9 0E
C808: 60 48 8A 48 98 48 AD 37 49
C810: CA D0 1B 20 C0 C9 B0 19 3B
C818: A9 7F 8D 00 DC AD 01 DC 37
C820: C9 EF D0 03 4C 6E C9 C9 FB
C828: DF D0 03 4C 7C C9 4C 4C 07
C830: FE 20 82 C9 A9 05 8D 73 4B
C838: C8 A9 00 8D 75 C8 A9 01 21
C840: 8D 7B C8 A9 39 8D A3 C8 EE
C848: 8D 45 C9 A9 CA 8D A4 C8 54
C850: 8D 46 C9 A2 03 A0 0A 18 56
C858: 20 F0 FF A9 12 20 D2 FF 18
C860: A0 00 B9 00 CA 20 D2 FF 78
C868: C8 C0 14 D0 F5 A9 92 20 29
C870: D2 FF A2 03 A0 00 18 20 C1
C878: F0 FF A9 01 C9 0D D0 0C C7
C880: 48 A9 14 8D 75 C8 A9 05 01
C888: 8D 73 C8 68 CD 38 CA B0 3C
C890: 36 09 40 20 D2 FF A9 2E DA
C898: 20 D2 FF A9 20 20 D2 FF 48
C8A0: A0 00 B9 39 CA C9 A0 F0 5A
C8A8: 08 20 D2 FF C8 C0 10 D0 0E
C8B0: F1 EE 73 C8 EE 7B C8 AD AE
C8B8: A3 C8 18 69 10 8D A3 C8 B0
C8C0: 90 B0 EE A4 C8 B0 AB A2 5D
C8C8: 15 A0 0A 18 20 F0 FF A0 52
C8D0: 00 B9 14 CA 20 D2 FF C8 25
C8D8: C0 14 D0 F5 E8 A0 0C 18 22
C8E0: 20 F0 FF A0 00 B9 28 CA 3F
C8E8: 20 D2 FF C8 C0 0F D0 F5 3B
C8F0: A9 00 85 C6 20 9F FF 20 C6
C8F8: E4 FF C9 00 F0 F6 C9 20 79
C900: D0 03 4C 6E C9 C9 41 90 F3
C908: EB 29 3F CD 38 CA B0 E4 C2
C910: A8 88 C0 00 F0 12 AD 45 F7
C918: C9 18 69 10 8D 45 C9 90 A0
C920: 03 EE 46 C9 88 4C 12 C9 D2
C928: A9 93 20 D2 FF 18 A2 02 15
C930: A0 00 20 F0 FF A0 00 B9 3C
C938: F8 C9 20 D2 FF C8 C0 05 7C
C940: D0 F5 A0 00 B9 39 CA C9 2F
C948: A0 F0 08 20 D2 FF C8 C0 5E
C950: 10 D0 F1 A0 00 B9 FD C9 45
C958: 20 D2 FF C8 C0 03 D0 F5 9E
C960: 18 A2 07 20 9A C9 8D 78 AC
C968: 02 A9 02 4C B0 C9 20 82 7F
C970: C9 18 A2 02 20 9A C9 A9 25
C978: 01 4C B0 C9 20 82 C9 4C F8
C980: B7 C9 A9 01 8D 37 CA 20 5C
C988: 8A FF 20 EA C9 20 A3 FD A9
C990: 20 81 FF A9 00 85 C7 85 AE
C998: D4 60 A0 00 20 F0 FF B9 39
C9A0: F5 C9 20 D2 FF C8 C0 03 DF
C9A8: D0 F5 A9 0D 8D 77 02 60 8D
C9B0: 85 C6 A9 13 20 D2 FF A9 56
C9B8: 00 8D 37 CA 58 6C 02 A0 AF
C9C0: AD 01 DC CD 01 DC D0 F8 C1
C9C8: A9 BF 8D 00 DC AD 01 DC 28
C9D0: C9 EF F0 0F A9 FD 8D 00 BF
C9D8: DC AD 01 DC C9 7F F0 03 7E
C9E0: 18 90 01 38 A9 7F 8D 00 79
C9E8: DC 60 A9 09 8D 18 03 A9 2B
C9F0: C8 8D 19 03 60 52 55 4E B9
C9F8: 4C 4F 41 44 22 22 2C 38 C2
CA00: 53 48 4F 54 47 55 4E 20 4A
CA08: 4C 4F 41 44 20 41 4E 44 1D
CA10: 20 52 55 4E 50 52 45 53 61
CA18: 53 20 53 50 41 43 45 20 19
CA20: 54 4F 20 52 45 52 55 4E 71
CA28: 43 55 52 52 45 4E 54 20 6D
CA30: 50 52 4F 47 52 41 4D 4A
```

*This program must be entered using *Flankspeed*! See page 118.

IMPORTANT! Letters on white background are **Bug Repellent** line codes. **Do not enter them!** Pages 117 and 118 explain these codes and provide other essential information on entering *Ahoy!* programs. Refer to these pages **before** entering any programs!

DISK ERROR CHECKER

FROM PAGE 59

```
•100 REM GEORGE TREPAL COPYRIGHT 1985        FM
•110 REM 2650 ALTURAS RD                     PO
•120 REM BARTOW, FL   33830                  AG
•130 CLR:POKE53281,0:POKE53280,0:DIMTR(40
 ):PRINT"[CLEAR][c 6][DOWN]"                 JB
•140 PRINTSPC(11)"[WHITE]DISK ERROR CHECK
 ER"                                         BA
•150 PRINTSPC(8)"[DOWN][YELLOW]NO KNOCK -
 - TRACKS 1-40"                              CE
•160 PRINTSPC(12)"[CYAN][DOWN]BY GEORGE T
 REPAL"                                      OM
•170 PRINT"[DOWN][DOWN][c 6]CHECK <A>LL T
 RACKS OR <S>ELECTED TRACKS"                 EK
•180 GETY$:IFY$<>"A"ANDY$<>"S"THEN180        LN
•190 IFY$="A"THENFT=1:LT=40:GOTO260          HF
•200 FT=-1:LT=50                             CC
•210 INPUT"[DOWN][DOWN][8" "]FIRST TRACK
 TO CHECK";FT                                BG
•220 IFFT<1ORFT>40THEN210                    PP
•230 INPUT"[DOWN][8" "]LAST TRACK TO CHEC
 K";LT                                       ON
•240 IFLT<1ORLT>40THEN230                    KJ
•250 IFLT<FTTHEN210                          MI
•260 PRINTSPC(5)"[DOWN]PRINT OUT ERRORS F
 OUND (Y/N)?"                                GI
•270 GETY$:IFY$<>""THEN270                   EB
•280 PF=0:GETY$:IFY$=""THEN280               GA
•290 IFY$="Y"THENPF=1                        JK
•300 PRINT"[DOWN][DOWN] INSERT DISK TO CH
 ECK THEN PRESS RETURN"                      PK
•310 PRINT SPC(2)"(FLASHING [c 3]RED[c 6]
  DRIVE LIGHT IS NORMAL)"                    KE
•320 GETA$:IFA$<>CHR$(13)THEN320             DG
•330 OPEN15,8,15:PRINT#15,"I":CLOSE15        DJ
•340 REM DEFINE # OF SECTORS ON TRACKS       NC
•350 FORJ=1TO17:TR(J)=20:NEXT:FORJ=18TO23
 :TR(J)=18:NEXT                              MK
•360 FORJ=24TO30:TR(J)=17:NEXT:FORJ=31TO4
 0:TR(J)=16:NEXT                             LH
•370 FORT=FTTOLT:PRINT"[CLEAR][WHITE]";:P
 RINTSPC(15)"TRACK ";T                       OE
•380 PRINT"[c 6][DOWN][3" "]SECTOR[4" "]E
 RROR[5" "]EXPLANATION"                      EA
•390 FORS=0TOTR(T)                           KB
•400 PRINT"[c 6]SECTOR # ";S;:IFS<10THENP
 RINT" ";                                    LL
•410 OPEN15,8,15                             AM
•420 PRINT#15,"M-W"CHR$(6)CHR$(0)CHR$(2)C
 HR$(T)CHR$(S)                               KO
•430 PRINT#15,"M-W"CHR$(0)CHR$(0)CHR$(1)C
 HR$(128)                                    LM
•440 PRINT#15,"M-R"CHR$(0)CHR$(0)            PE
•460 IFA=1THENPRINT" NO ERRORS":GOTO550      EH
•470 IFA=2THENB$="[YELLOW]BLOCK HEADER GO
 NE"                                         EH
•480 IFA=3THENB$="[CYAN]SYNC CHARACTER GO
 NE"                                         JG
•490 IFA=4THENB$="[c 7]HEADER CHECKSUM BA
 D"                                          HM
•500 IFA=5THENB$="[PURPLE]READ CHECKSUM B
 AD"                                         NB
•510 IFA=6THENB$="[BLUE]BIT PATTERN ERROR
 "                                           AL
•520 IFA=9THENB$="[WHITE]HEADER CHECKSUM
 BAD"                                        LD
•530 IFA=11THENB$="[c 1]FORMAT ID MISMATC
 H"                                          HA
•540 PRINT" ";A+18;" ";B$                    BD
•550 CLOSE15                                 AB
•560 IFPF=0ORA=1THEN590                      CI
•570 OPEN4,4:CMD4:PRINT#4,"TRACK ";T;"[3"
 "]SECTOR";S;"  ERROR #";A+18;               GC
•580 PRINT#4,"[4" "]";B$:CLOSE4              IM
•590 NEXT:NEXT                               EF
•600 PRINTSPC(8)"[WHITE]PRESS RETURN TO C
 ONTINUE[c 6]"                               JE
•610 GETY$:IFY$<>CHR$(13)THEN610             KI
•620 GOTO130                                 CA
```

MAESTRO

FROM PAGE 39

```
•3 POKE56334,PEEK(56334)AND254               DN
•4 FORX=36609TO40959:POKEX,Z:NEXT            KH
•6 POKE55,4:POKE56,109:CLR:GOTO10            FG
•9 N%(K,O)=FN H(FO):N%(K,Z)=FN L(FO):FO=F
 O*R:RETURN                                  GI
•10 POKE53280,0:POKE53281,1:POKE646,0:PRI
 NT"[CLEAR]":GOSUB2000:GOSUB3000             MM
•14 Q=54272:R=1.0594631:FO=43.65:SK=65439
 :DO=53248:KY=197                            FK
•15 SH=653:F1=.064068:F2=16.4015:F3=256:O
 =1:W=16:JI(0)=-1:JI(1)=-1:JI(2)=-1:T=2      KD
•16 Q1=Q+7:Q2=Q+14:Q3=Q+0:Q4=Q+8:Q5=Q+15:
 Q6=Q+4:Q7=Q+11:Q8=Q+18:E1=Q6:E2=Q7:E3=Q8    HB
•17 TH=119:FH=401:SI=1450:C$=CHR$(13):W1=
 W:W2=W:W3=W:SR=240                          GB
•20 L1=27909:L2=29359:L3=30809:H1=32259:H
 2=33709:H3=35159:O1=36609:O2=38059          CC
•22 O3=39509:OV=0:HV=0:LV=0                  HE
•28 DIMM%(400,2):DIMR%(400,2)                EB
•30 DIMN%(64,1):DIMCH%(64)                   CK
•40 DEF FN H(FO)=INT(FO*F1)                  MA
•50 DEF FN L(FO)=INT(FO*F2)-(F3*INT(FO*F1
 ))                                          OL
•55 FORI=ZTO64:READJ,K:CH%(I)=J:IFJ>ZANDJ
 <>THTHENGOSUB9                              DD
•58 NEXT                                     IA
```

```
•60 FORI=QTO54295:POKEI,Z:NEXT:POKEI,15 KN
•70 POKEQ+6,SR:POKEQ+13,SR:POKEQ+20,SR PB
•71 FORX=0TO9:READT$(X):NEXT:GOTO608 JI
•140 OV=O1+V*SI CJ
•142 FORX=ZTOK(V):POKEOV+X,Z:NEXT:K(V)=-1:JI(V)=-1:JJ=Z:KK=Z:C=Z:K(3)=Z:RETURN IK
•150 GETA$:IFA$<>""THEN150 AO
•160 PRINT"[CLEAR][DOWN][DOWN][RIGHT]GET READY TO PLAY [3"."]";:FORX=OTOFH:NEXT:PRINT"GO!" BA
•170 POKEQ+5,AD:POKEQ+6,SR HO
•295 POKEQ6,W+O:U=49152 BN
•300 SYSSK:IFPEEK(SH)=OTHEN360 BN
•305 IFU=DOTHEN363 HA
•310 X=PEEK(KY):POKEQ,N%(X,Z):POKEQ3,N%(X,1):POKEU,N%(X,Z):POKEU+1,N%(X,O) MB
•320 U=U+T:GOTO300 LI
•360 IFSTHEN365 CB
•362 RETURN IM
•363 IFSTHEN365 CB
•364 GOTO295 CK
•365 PRINT"[CLEAR][DOWN][DOWN][RIGHT]HERE'S WHAT YOU JUST PLAYED:" EJ
•372 FORI=49152TOU-TSTEPT KI
•373 SYSSK:IFPEEK(SH)=TTHEN381 AB
•375 POKEQ,PEEK(I):POKEQ3,PEEK(I+O) BI
•380 FORJ=OTO35:NEXT:NEXT EP
•381 POKEQ6,W:RETURN HO
•386 PRINT"[DOWN][DOWN][RIGHT]NOTES:  "; GO
•390 SYSSK:IFPEEK(SH)=OTHENRETURN PP
•400 GETA$:IFA$=""THEN390 HD
•401 IFPEEK(SH)=TTHEN411 IF
•402 X=PEEK(KY):PRINTCHR$(CH%(X)); HH
•405 POKEQ,N%(X,Z):POKEQ3,N%(X,O):POKEQ6,W+1:FORI=OTOF3:NEXT:POKEQ6,W BJ
•410 M%(JJ,V)=X:JJ=JJ+O:IFJJ<FHTHEN390 MD
•411 IFJJ>CTHENJJ=JJ-O:PRINT"[LEFT] [LEFT]"; BN
•412 GOTO390 CE
•420 IFJI(V)<ZTHENFORX=OTO999:NEXT:K(V)=-1:RETURN EA
•421 PRINT:PRINT" RHYTHM: "; HM
•422 SYSSK:GETA$:IFA$=""THEN422 NJ
•425 IFPEEK(SH)=TTHEN432 IC
•426 IFA$<"A"ORA$>"Z"THEN422 NL
•430 PRINTA$;:R%(KK,V)=ASC(A$)-64:KK=KK+O:IFKK<=JI(V)THEN422 KG
•431 GOTO435 CK
•432 IFKK>CTHENKK=KK-O:PRINT"[LEFT] [LEFT]"; BN
•434 GOTO422 CI
•435 K=K(V)+O:HV=H1+V*SI:LV=L1+V*SI:OV=O1+V*SI AB
•436 FORJ=CTOJI(V) LG
•437 FORX=OTOR%(J,V)-O:POKEHV+K,N%(M%(J,V),O):POKELV+K,N%(M%(J,V),Z) OH
•438 POKEOV+K,O:K=K+O:IFK=SITHENRETURN JM
•439 NEXT IA
•440 IFR%(J,V)<=OTHEN450 BM
•444 POKEHV+K,N%(M%(J,V),O):POKELV+K,N%(M%(J,V),Z):POKEOV+K,Z:K=K+O DL
•450 NEXT:RETURN EJ
•470 FORX=OTO999:NEXT:GOSUB2000 HH
•520 FORI=ZTOK(S-O) HG
•525 SYSSK:IFPEEK(SH)=OTHEN600 AG
•530 POKEQ,PEEK(L1+I):POKEQ1,PEEK(L2+I):POKEQ2,PEEK(L3+I) KN
•540 POKEQ3,PEEK(H1+I):POKEQ4,PEEK(H2+I):POKEQ5,PEEK(H3+I) NF
•550 POKEE1,PEEK(O1+I)+W1:POKEE2,PEEK(O2+I)+W2:POKEE3,PEEK(O3+I)+W3 IP
•560 FORX=ZTOE:NEXT:NEXT EE
•600 POKEQ6,Z:POKEQ7,Z:POKEQ8,Z:FORX=ZTOF3:NEXT:RETURN CP
•608 GETA$:IFA$<>""THEN608 AA
•609 PRINT"[CLEAR]":POKE53280,6:POKE53281,1:PRINTTAB(5)"[DOWN][RED][3"*"]  S E L E C T I O N S  [3"*"]" OK
•610 PRINTTAB(10)"[BLUE][3"[DOWN]"]0-MUSICAL KEYBOARD" CL
•615 PRINTTAB(10)"1-PLAY IT AGAIN!" AO
•620 PRINTTAB(10)"2-COMPOSE A TUNE" JF
•625 PRINTTAB(10)"3-LOOK AT MUSIC" FL
•630 PRINTTAB(10)"4-MUSIC EDITOR" EK
•632 PRINTTAB(10)"5-SELECT INSTRUMENT" PI
•635 PRINTTAB(10)"6-LISTEN TO MUSIC" LN
•640 PRINTTAB(10)"7-SAVE MUSIC ON DISK" GK
•642 PRINTTAB(10)"8-GET MUSIC FROM DISK" NJ
•644 PRINTTAB(10)"9-LOOK AT DISK INDEX" BP
•645 PRINTTAB(10)"X-QUIT PROGRAM" CD
•646 PRINTTAB(10)"[RED][DOWN][DOWN]YOUR CHOICE [0-X]?"; BG
•650 SYSSK:GETA$:IFA$=""THEN650 NI
•652 IFA$="X"THEN1350 KF
•654 S=VAL(A$):IFS=0ANDA$<>"0"THEN650 LH
•660 ON-(S=0)-2*(S=1)-3*(S=2)-4*(S=3)-5*(S=4)GOSUB150,150,676,710,800 NJ
•665 ON-(S=5)-2*(S=6)-3*(S=7)-4*(S=8)-5*(S=9)GOSUB1150,900,1000,1050,1100 AI
•670 GOTO608 CK
•676 PRINT"[CLEAR][DOWN][DOWN] HOW MANY VOICES (1,2,3) ?"; PI
•678 SYSSK:IFPEEK(SH)THENRETURN PJ
•679 GETA$:IFA$=""THEN678 KA
•681 B=VAL(A$):IFB>3ORB=ZTHEN678 MB
•682 FORV=ZTOT:GOSUB140:NEXT PP
•685 FORV=ZTOB-O:PRINT"[CLEAR][DOWN][DOWN] INPUT VOICE"V+O NC
•686 JJ=Z:KK=Z JG
•688 GOSUB386:JI(V)=JJ-O:GOSUB420:K(V)=K-O:S=V+O:GOSUB920:NEXT:RETURN BL
•710 PRINT"[CLEAR][DOWN][DOWN] WHICH VOICE (1,2,3) ?"; OC
•712 SYSSK:IFPEEK(SH)=OTHENRETURN PP
•715 GETA$:IFA$=""THEN712 KB
•718 V1=VAL(A$):IFV1>3ORV1=ZTHEN712 HN
```

```
•719 V=V1-O:IFJI(V)>=ZTHEN721                    FH
•720 PRINT"[CLEAR][DOWN] NO MUSIC ASSIGNE
 D TO VOICE";V1:FORX=OTO2000:NEXT:RETURN   LM
•721 PRINT"[CLEAR][DOWN][DOWN][RIGHT]VOIC
 E"V1:PRINT"[DOWN][DOWN][RIGHT]NOTES:  ";
 :FORJ=ZTOJI(V)                               FL
•724 PRINTCHR$(CH%(M%(J,V)));:NEXT:PRINT:
 PRINT                                        ND
•735 PRINT"[DOWN] RHYTHM: ";:FORJ=ZTOJI(V
 ):PRINTCHR$(R%(J,V)+64);:NEXT                PM
•739 SYSSK:IFPEEK(SH)=OTHENRETURN             PP
•740 GOTO739                                  DB
•800 PRINT"[CLEAR][RIGHT][DOWN][DOWN]EDIT
  WHICH VOICE (1,2,3) ?";                     EK
•803 SYSSK:IFPEEK(SH)=1THENRETURN             MB
•805 GETA$:IFA$=""THEN803                     KA
•806 V1=VAL(A$):IFV1>3ORV1=ZTHEN805           GI
•808 V=V1-O:IFJI(V)>=ZTHEN810                 FE
•809 PRINT"[CLEAR][DOWN] NO MUSIC ASSIGNE
 D TO VOICE";V1:FORX=OTO2000:NEXT:GOTO823  EP
•810 PRINT"[CLEAR][DOWN][DOWN][RIGHT]THE
 NOTES CURRENTLY ARE:":PRINT"[DOWN] NOTES
 :  ";                                        FH
•816 FORJ=ZTOJI(V):PRINTCHR$(CH%(M%(J,V))
 );:NEXT:PRINT                                AH
•817 PRINT"[DOWN] IF OK TYPE <SPACE> ":PR
 INT" 'B' TO START FROM BEGINNING"            HE
•818 PRINT" 'A' TO CONTINUE ADDING NOTES"  DC
•819 SYSSK:GETA$:IFA$=""THEN819               OB
•820 IFA$="A"THEN850                          GA
•821 IFA$=" "THENTJI=JI(V):GOTO826            ME
•822 IFA$<>"B"THEN817                         CA
•823 PRINT"[DOWN][DOWN] INPUT THE NOTES A
 S THEY SHOULD BE."                           FL
•824 PRINT:GOSUB140:GOSUB386:TJI=JJ-O         JF
•825 IFJI(V)<ZTHEN840                         EG
•826 PRINT:PRINT"[DOWN] THE RHYTHM CURREN
 TLY IS":PRINT:PRINT"[DOWN] RHYTHM: ";        CP
•827 FORJ=ZTOJI(V):PRINTCHR$(R%(J,V)+64);
 :NEXT:IFTJI<>JI(V)THEN840                    FJ
•832 PRINT:PRINT"[DOWN][DOWN] IF OK TYPE
 <SPACE> ":PRINT" OTHERWISE 'R'"              EH
•833 GETA$:IFA$<>""THEN833                    AA
•834 SYSSK:GETA$:IFA$=""THEN834               NM
•835 IFA$=" "THENK(V)=-1:C=Z:KK=Z:GOSUB43
 5:K(V)=K-O:RETURN                            EM
•840 JI(V)=TJI:PRINT:PRINT"[DOWN][DOWN] I
 NPUT THE RHYTHM AS IT SHOULD BE."            FL
•842 C=Z:KK=Z:K(V)=-1:GOSUB420                HM
•845 K(V)=K-O:RETURN                          MJ
•850 JJ=JI(V)+O:KK=JJ:C=JJ:GOSUB386:JI(V)
 =JJ-O                                        MK
•860 IFJJ>KKTHENGOSUB420:K(V)=K-O             FC
•870 S=V+O:GOSUB920:RETURN                    IB
•900 PRINT"[CLEAR][DOWN][DOWN] TYPE NUMBE
 R OF VOICE TO BE PLAYED ";                   AM
•910 PRINT:PRINT"[DOWN] OR TYPE <4> FOR A
 LL VOICES AT ONCE "                          BB
•912 SYSSK:IFPEEK(SH)THENRETURN               PJ
•914 GETA$:IFA$=""THEN912                     IL
•916 S=VAL(A$):IFS>4ORS=ZTHEN912              MA
•918 PRINTS:IFS<4ANDJI(S-O)=-1THENV1=S:GO
 TO720                                        DD
•920 E1=-1*(S=4ORS=O)*(Q6-820)+820:E2=-1*
 (S=4ORS=T)*(Q7-820)+820                      NN
•924 E3=-1*(S=4ORS=3)*(Q8-820)+820:K(3)=K
 (Z):IFK(O)>K(3)THENK(3)=K(O)                 BG
•928 IFK(2)>K(3)THENK(3)=K(2)                 JE
•935 PRINT"[CLEAR][DOWN][DOWN] INDICATE T
 EMPO (0-9) "                                 DA
•938 SYSSK                                    DL
•940 GETA$:IFA$=""THEN938                     IL
•945 E=VAL(A$)*10:IFE=ZANDA$<>"0"THEN935   LO
•965 GOSUB470:RETURN                          OO
•1000 POKE56334,PEEK(56334)OR1:A$="":B$="
 "                                            PC
•1005 PRINT"[CLEAR][DOWN][DOWN] INSERT FO
 RMATTED DISKETTE"                            GA
•1010 INPUT"[DOWN][DOWN] NAME YOUR PIECE"
 ;A$:IFA$=""THEN1042                          KD
•1016 INPUT"[DOWN] RECOMMENDED TEMPO";B$:
 IFB$=""THEN1042                              MD
•1018 E=VAL(B$):IFE=ZANDB$<>"0"THEN1016    JN
•1020 OPEN15,8,15,"I0":OPEN5,8,5,"@0:"+A$
 +",S,W"                                      DN
•1030 PRINT#5,E;C$;JI(Z);C$;JI(O);C$;JI(T
 )                                            LJ
•1036 FORV=ZTOT:FORX=ZTOJI(V):PRINT#5,M%(
 X,V);C$;R%(X,V):NEXT:NEXT                    MK
•1040 CLOSE5:CLOSE15                          BF
•1041 PRINT" "A$" SAVED ON DISK"              DP
•1042 POKE56334,PEEK(56334)AND254:FORX=ZT
 O999:NEXT:RETURN                             EB
•1050 FORV=ZTOT:GOSUB140:NEXT                 PP
•1052 POKE56334,PEEK(56334)ORO:A$="":INPU
 T"[CLEAR][DOWN][DOWN] NAME OF PIECE";A$   NC
•1054 OPEN15,8,15,"I0":OPEN5,8,5,"0:"+A$+
 ",S,R":INPUT#5,E,JI(Z),JI(O),JI(T)           JI
•1060 FORV=ZTOT:FORX=ZTOJI(V):INPUT#5,M%(
 X,V),R%(X,V):NEXT:NEXT                       ID
•1062 POKE56334,PEEK(56334)AND254             DN
•1065 INPUT#15,E$                             ID
•1066 IFVAL(E$)>ZTHENPRINT"[DOWN][DOWN] "
 A$" NOT ON DISK":FORX=ZTO999:NEXT:GOTO 1
 074                                          BN
•1067 PRINT"[DOWN] "A$" LOADED FROM DISK"
 :C=Z:PRINT" RECOMMENDED TEMPO:";E            PN
•1068 FORV=ZTOT:K(V)=-1                       JN
•1070 PRINT" COMPILING VOICE"V+1:IFJI(V)>
 -1THENGOSUB435:K(V)=K-O                      CN
•1073 NEXT                                    IA
•1074 CLOSE5:CLOSE15:RETURN                   NI
•1100 POKE56334,PEEK(56334)OR1:PRINT"[CLE
 AR][DOWN][DOWN] THIS DISK CONTAINS:":PRI
 NT                                           GB
•1110 OPEN15,8,15,"I0":OPEN5,8,5,"$"          AE
```

```
•1114 FORJ=1TO254:GET#5,A$:NEXT             FB
•1117 FORX=0TO8:GET#5,D$,A$,A$:IFSTTHEN11
45                                          NC
•1130 B$="":FORJ=0TO16:GET#5,A$:IFA$<>CHR
$(160)THENB$=B$+A$                          AI
•1133 NEXT:IFASC(D$+CHR$(Z))=129THENPRINT
,B$                                         PK
•1140 FORJ=1TO11:GET#5,A$:NEXT:IFX<8THENG
ET#5,A$,A$                                  HD
•1142 NEXT:GOTO1117                         BA
•1145 CLOSE5:CLOSE15:A$="":INPUT"[DOWN] <
RETURN> TO RETURN";A$                       NE
•1146 POKE56334,PEEK(56334)AND254:RETURN    PM
•1150 PRINT"[CLEAR]":FORX=QTOQ+23:POKEX,Z
:NEXT                                       CF
•1151 FORX=ZTOT:IFJI(X)<ZANDXTHEN1313       OL
•1159 PRINT"[DOWN][DOWN] CHOOSE DESIRED S
OUND FOR VOICE:";X+O                        JL
•1160 PRINT"[DOWN][DOWN] PRESS A NUMBER K
EY [3"."] ";                                MC
•1161 SYSSK:IFPEEK(SH)THEN1313              DF
•1162 GETA$:IFA$=""THEN1161                 KF
•1163 S=VAL(A$):IFS=0ANDA$<>"0"THEN1161     AA
•1164 PRINTT$(S):FORJ=ZTOFH:NEXT            IH
•1166 ON-(S=0)-2*(S=1)-3*(S=2)-4*(S=3)-5*
(S=4)GOTO1200,1210,1220,1230,1240           JL
•1167 ON-(S=5)-2*(S=6)-3*(S=7)-4*(S=8)-5*
(S=9)GOTO1250,1260,1270,1280,1290           HJ
•1200 W=16:AD=144:SR=243:GOTO1300           FG
•1210 W=16:AD=0:SR=240:GOTO1300             OF
•1220 W=16:AD=88:SR=195:GOTO1300            DL
•1230 W=32:AD=88:SR=95:GOTO1300             DH
•1240 W=32:AD=0:SR=240:GOTO1300             NP
•1250 W=128:AD=88:SR=195:GOTO1300           NF
•1260 W=64:AD=14:SR=86:HP=6:LP=2:GOTO1300   HC
•1270 W=64:AD=9:SR=90:HP=8:LP=8:GOTO1300    AB
•1280 W=64:AD=9:SR=90:HP=0:LP=255:GOTO130
0                                           EK
•1290 W=16:AD=9:SR=80                       BM
•1300 IFX=ZTHENW1=W                         DD
•1310 IFX=OTHENW2=W                         CH
•1311 IFX=TTHENW3=W                         CL
•1312 J=Q+T+7*X:POKEJ,LP:POKEJ+O,HP:POKEJ
+3,AD:POKEJ+4,SR                            ED
•1313 NEXT:S=10:RETURN                      PB
•1350 POKE56334,PEEK(56334)OR1:END          II
•1400 DATA 97,12,120,23,62,20,35,31,36,28
,37,39,38,36,60,47,51,44,87,55              PD
•1410 DATA 65,7,52,2,90,3,83,63,69,10,0,0
,53,13,82,18,68,21,54,26,67,29              OB
•1500 DATA 70,34,84,37,88,42,55,45,89,50,
71,53,56,1,66,6,72,62,85,9,86,14            CH
•1510 DATA 57,17,73,22,74,25,48,30,77,33,
75,38,79,41,78,46,43,49,80,54,76,5,45,57    HI
•1600 DATA 46,56,58,59,64,8,44,11,92,16,4
2,19,59,24,115,27,0,0,61,32                 OF
•1610 DATA 94,35,47,40,49,43,95,48,0,0,50
,51,119,0,0,0,81,0,126,4,0,0                LO
•1700 DATAACCORDIAN,ORGAN,CLARINET,SYNTHE
SIZER,KAZOO,DRUM,GUITAR,XYLOPHONE           FG
•1710 DATABANJO,PIANO                       LD
•2000 PRINT "[CLEAR][3"[DOWN]"][BLUE][11"
"][s U][s I]"                               AD
•2001 PRINT "[11" "][s G][s K]"             NP
•2002 PRINT "[3" "][8"[c T]"][s N][c T]     BM
•2003 PRINT "[3" "][7"[c T]"][s N][s G][c
T]                                          JN
•2004 PRINT "[3" "][6"[c T]"][c M][s U][s
G][s I][5"[c T]"][RED]M A E S T R O[BLU
E][5"[c T]"]"                               BI
•2005 PRINT "[3" "][6"[c T]"][c M][s J][s
G][c N][23"[c T]"]"                         HA
•2006 PRINT "[3" "][7"[c T]"][s M][s G][s
N][23"[c T]"]"                              JD
•2007 PRINT "[11" "][s G]"                  BE
•2008 RETURN                                IM
•3000 PRINT"                                LP
•3001 PRINT"[27" "][c M][3"[c Y]"][s P]     FG
•3002 PRINT"[PURPLE][4" "][s Q] ADVANCED[
BLUE][13" "][c M]  [c Y][s P]               AF
•3003 PRINT"[27" "][s Q].  [s Q]            IJ
•3004 PRINT"[GREEN][6" "][s Q] MUSIC[BLUE
]                                           AH
•3005 PRINT"                                LP
•3006 PRINT"[c 2][8" "][s Q] EDITOR[BLUE]
[5" "][c M]  [c M]  [c M]                   EG
•3007 PRINT"[21" "][c M]  [c M]  [c M]      LA
•3008 PRINT"[21" "][s Q]  [s Q]  [s Q]      DJ
•3009 PRINT:PRINTTAB(8)"BY WILLIAM T. EVE
LETH AND"                                   OM
•3010 PRINTTAB(12)"THE AHOY! STAFF"         MI
•3011 RETURN                                IM
```

PROGRAMMING BIT BY BIT

FROM PAGE 103 THE NAME GAME

```
*
* THE NAME GAME
*
        ORG     $8000
*
EOL     EQU     $0D         ;RETURN
EOF     EQU     $03         ;EOF CHR
FILLCH  EQU     $20         ;SPACE
BUFLEN  EQU     40
CHRIN   EQU     $FFCF
CHROUT  EQU     $FFD2
TEMPTR  EQU     $FB
*
        JMP     START
*
TXTBUF  DS      40
```

```
*
TITLE    ASC   'THE NAME GAME'
         HEX   0D
HELLO    ASC   'HELLO, '
         HEX   03
QUERY    ASC   'WHAT IS YOUR NAME?'
         HEX   0D
NAME     ASC   'GEORGE'
         HEX   0D
REBUFF   ASC   'GO AWAY, '
         HEX   03
DEMAND   ASC   'BRING ME GEORGE!'
         HEX   0D
GREET    ASC   'HI, GEORGE!'
         HEX   0D
*
* CLEAR TEXT BUFFER
*
FILL     LDA   #FILLCH
         LDX   #BUFLEN
DOFILL   DEX
         STA   TXTBUF,X
         BNE   DOFILL
         RTS
*
PRINT    LDY   #0
SHOW     LDA   (TEMPTR),Y
         CMP   #EOF
         BEQ   DONE
         PHA
         JSR   CHROUT
         PLA
         CMP   #EOL
         BNE   NEXT
         JMP   DONE
NEXT     INY
         CPY   #BUFLEN
         BCC   SHOW
DONE     RTS
*
* PRINT 'THE NAME GAME'
*
START    LDA   #EOL
         JSR   CHROUT
         LDA   #<TITLE
         STA   TEMPTR
         LDA   #>TITLE
         STA   TEMPTR+1
         JSR   PRINT
         LDA   #EOL
         JSR   CHROUT

*
* PRINT 'HELLO . . .'
*
         LDA   #<HELLO
         STA   TEMPTR
         LDA   #>HELLO
         STA   TEMPTR+1
         JSR   PRINT
*
* PRINT 'WHAT IS YOUR NAME?'
*
ASK      LDA   #<QUERY
         STA   TEMPTR
         LDA   #>HELLO
         STA   TEMPTR+1
         JSR   PRINT
         LDA   #EOL
         JSR   CHROUT
*
* INPUT A TYPED LINE
*
         JSR   FILL
         LDX   #0
KEY      JSR   CHRIN
         STA   TXTBUF,X
         CMP   #EOL
         BEQ   COMPARE
         INX
         JMP   KEY
*
* IS THE NAME 'GEORGE'?
*
COMPARE  JSR   CHROUT       ;PRINT RETURN
         LDX   #0
CHECK    LDA   TXTBUF,X
         CMP   NAME,X
         BNE   NOGOOD
         CMP   #EOL
         BEQ   DUNIT
         INX
         CPX   #BUFLEN
         BCS   DUNIT
         JMP   CHECK
*
* NO; PRINT 'GO AWAY . . .'
*
NOGOOD   LDA   #EOL
         JSR   CHROUT
         LDA   #<REBUFF
         STA   TEMPTR
         LDA   #>REBUFF
         STA   TEMPTR+1
         JSR   PRINT
*
* PRINT PLAYER'S NAME
*
         LDA   #<TXTBUF
         STA   TEMPTR
         LDA   #>TXTBUF
         STA   TEMPTR+1
         JSR   PRINT
         LDA   #EOL
         JSR   CHROUT
*
```

IMPORTANT! Letters on white background are **Bug Repellent** line codes. **Do not enter them!** Pages 117 and 118 explain these codes and provide other essential information on entering *Ahoy!* programs. Refer to these pages **before** entering any programs!

```
* PRINT 'BRING ME GEORGE!'
*
         LDA   #<DEMAND
         STA   TEMPTR
         LDA   #>DEMAND
         STA   TEMPTR+1
         JSR   PRINT
         LDA   #EOL
         JSR   CHROUT
         JMP   ASK
*
* YES; PRINT GREETING
*
DUNIT    LDA   #EOL
         JSR   CHROUT
         LDA   #<GREET
         STA   TEMPTR
         LDA   #>GREET
         STA   TEMPTR+1
         JSR   PRINT
         RTS
```

SHADEYDUMP

FROM PAGE 93

```
•100 PRINT "[CLEAR]"CHR$(14)CHR$(8)SPC(13
 )"[s S]HADEY[s D]UMP"                      PK
•110 PRINT SPC(12)"[s V]ERSION  2.0"        LM
•120 PRINT SPC(45)"(C) 1985 BY [s M]ICHAE
 L [s S]CHELL"                              FJ
•130 FORA=49152 TO 50001:READ B:C=C+B:POK
 E A,B:NEXT                                 JO
•140 IF C<>86106 THEN PRINT "ERROR IN DAT
 A STATEMENTS":STOP                         FK
•150 PRINT SPC(120)"[s S]AVING MACHINE LA
 NGUAGE PROGRAM[4"."]"                      AM
•160 SYS 49872, 49152, 50002, "0:SHADEYDU
 MP.ML",8                                   FH
•170 PRINT "[HOME]"SPC(140)SPC(140)"[5" "
 ][s T]O RUN [s S]HADEY[s D]UMP ENTER:[6"
 "]"                                        ME
•180 PRINT SPC(5) "SYS 49152"               OL
•190 END                                    IC
•200 DATA032,104,192,162,000,134            PJ
•210 DATA091,162,001,134,093,032            NJ
•220 DATA203,192,162,000,134,094            BD
•230 DATA032,216,192,173,048,194            NP
•240 DATA240,003,162,024,044,162            MJ
•250 DATA000,134,095,032,035,193            AI
•260 DATA173,048,194,240,003,162            GB
•270 DATA007,044,162,000,134,096            ID
•280 DATA032,178,193,173,048,194            BN
•290 DATA240,012,166,096,202,016            AK
•300 DATA241,166,095,202,016,221            EG
•310 DATA048,014,166,096,232,224            GC
•320 DATA008,208,227,166,095,232            PF
•330 DATA224,025,208,205,166,094            PM
•340 DATA232,224,002,208,183,166            MM
•350 DATA093,202,016,171,166,091            CH
•360 DATA232,224,040,208,160,076            DH
•370 DATA237,192,173,049,194,240            BI
•380 DATA029,169,255,141,003,221            LL
•390 DATA173,038,003,141,050,194            NC
•400 DATA173,039,003,141,051,194            PM
•410 DATA169,190,141,038,003,169            HI
•420 DATA192,141,039,003,208,024            IB
•430 DATA169,004,174,052,194,172            DL
•440 DATA053,194,032,186,255,169            BG
•450 DATA000,032,189,255,032,192            KG
•460 DATA255,162,004,032,201,255            GB
•470 DATA032,185,192,169,051,032            AG
•480 DATA210,255,169,001,032,210            EI
•490 DATA255,169,013,032,210,255            GC
•500 DATA169,010,076,210,255,169            EN
•510 DATA027,076,210,255,072,141            BI
•520 DATA001,221,169,016,044,013            PH
•530 DATA221,240,251,104,096,032            MG
•540 DATA185,192,169,074,032,210            AP
•550 DATA255,169,013,076,172,192            DH
•560 DATA032,175,192,032,185,192            OH
•570 DATA169,121,032,210,255,169            DG
•580 DATA032,032,210,255,169,003            AI
•590 DATA076,210,255,173,049,194            PC
•600 DATA240,013,173,050,194,141            CO
•610 DATA038,003,173,051,194,141            NN
•620 DATA039,003,096,032,204,255            GB
•630 DATA169,004,076,195,255,169            CC
•640 DATA000,133,105,133,106,162            HA
•650 DATA008,010,038,106,006,107            FF
•660 DATA144,007,024,101,108,144            LA
•670 DATA002,230,106,202,208,239            MC
•680 DATA133,105,096,165,095,133            EN
•690 DATA107,169,040,133,108,032            PJ
•700 DATA007,193,165,091,024,101            AG
•710 DATA105,133,105,144,002,230            DH
•720 DATA106,165,105,133,087,165            DD
•730 DATA106,133,088,173,040,194            KB
•740 DATA024,101,105,133,105,133            HO
•750 DATA089,173,041,194,101,106            IO
•760 DATA133,106,133,090,160,000            LL
•770 DATA177,089,041,015,133,252            MP
•780 DATA177,089,074,074,074,074            BG
•790 DATA133,253,165,088,133,106            MB
•800 DATA165,087,133,105,024,109            NI
•810 DATA044,194,133,105,165,088            AH
•820 DATA109,045,194,133,106,160            LO
•830 DATA000,177,105,041,015,133            GP
•840 DATA251,173,046,194,133,105            IC
•850 DATA173,047,194,133,106,177            IP
•860 DATA105,041,015,133,255,165            PO
•870 DATA088,133,106,165,087,133            NO
•880 DATA105,162,003,006,105,038            HN
```

IMPORTANT! Letters on white background are **Bug Repellent** line codes. **Do not enter them!** Pages 117 and 118 explain these codes and provide other essential information on entering ***Ahoy!*** programs. Refer to these pages **before** entering any programs!

```
•890 DATA106,202,208,249,173,042       CD
•900 DATA194,024,101,105,133,105       MD
•910 DATA173,043,194,101,106,133       IC
•920 DATA106,096,164,096,177,105       CH
•930 DATA133,107,165,093,208,007       MA
•940 DATA162,004,006,107,202,208       BB
•950 DATA251,160,003,132,092,173       BG
•960 DATA039,194,240,015,160,004       AL
•970 DATA165,107,133,108,006,108       IH
•980 DATA169,000,042,170,232,208       HH
•990 DATA017,160,002,165,107,133       EO
•1000 DATA108,006,108,169,000,042      KG
•1010 DATA006,108,042,073,003,170      JE
•1020 DATA181,251,170,189,054,194      NP
•1030 DATA010,024,101,094,010,010      FM
•1040 DATA101,092,170,173,039,194      DK
•1050 DATA208,009,189,070,194,074      DP
•1060 DATA038,109,074,038,109,189      CL
•1070 DATA070,194,074,038,109,074      BI
•1080 DATA038,109,136,240,007,173      OH
•1090 DATA039,194,240,199,208,182      MK
•1100 DATA165,109,032,210,255,164      DJ
•1110 DATA092,136,016,159,096,255      PI
•1120 DATA000,092,000,096,000,128      CD
•1130 DATA033,208,255,000,000,000      GC
•1140 DATA004,005,015,000,011,004      PJ
•1150 DATA010,006,014,001,009,013      HO
•1160 DATA005,012,007,002,008,003      AI
•1170 DATA000,000,000,000,000,000      FG
•1180 DATA000,000,000,000,001,000      FN
•1190 DATA000,002,000,000,001,000      HH
•1200 DATA000,000,000,002,000,000      KA
•1210 DATA000,001,002,000,001,002      GA
•1220 DATA000,001,002,001,000,002      EE
•1230 DATA000,002,001,000,000,003      FA
•1240 DATA000,000,003,000,000,000      FP
•1250 DATA003,000,000,000,000,003      GA
•1260 DATA000,000,001,000,002,000      OH
•1270 DATA002,000,002,000,000,002      HE
•1280 DATA000,002,000,002,000,001      FF
•1290 DATA001,000,002,001,002,001      DD
•1300 DATA000,002,001,002,000,001      NI
•1310 DATA002,000,001,002,003,000      HO
•1320 DATA000,000,000,000,003,000      FP
•1330 DATA000,003,000,000,000,000      NL
•1340 DATA000,003,002,001,002,001      FB
•1350 DATA000,000,000,000,003,003      GA
•1360 DATA003,003,000,000,000,000      MA
•1370 DATA003,003,003,003,003,003      HE
•1380 DATA003,003,032,253,174,032      CK
•1390 DATA138,173,032,247,183,165      CP
•1400 DATA020,166,021,096,032,198      IM
•1410 DATA194,072,138,072,032,198      AN
•1420 DATA194,072,138,072,032,253      AL
•1430 DATA174,032,212,225,104,168      PD
•1440 DATA104,170,104,133,021,104      EL
•1450 DATA133,020,169,020,076,095      OC
•1460 DATA225,032,198,194,072,138      EP
•1470 DATA072,032,253,174,032,212      GO
•1480 DATA225,104,168,104,170,169      OE
•1490 DATA000,133,010,076,117,225      FL
•1500 DATA000,242,255,015,032,198      KD
•1510 DATA194,141,042,194,142,043      BE
•1520 DATA194,032,198,194,141,040      IN
•1530 DATA194,142,041,194,169,255      FE
•1540 DATA141,039,194,096,032,198      JA
•1550 DATA194,141,042,194,142,043      BE
•1560 DATA194,032,198,194,141,040      IN
•1570 DATA194,142,041,194,032,198      OP
•1580 DATA194,141,044,194,142,045      DI
•1590 DATA194,032,198,194,141,046      JD
•1600 DATA194,142,047,194,169,000      LI
•1610 DATA141,039,194,096              MB
```

1541 DISK DRIVE CLEANING UTILITY

FROM PAGE 59

```
•1 REM  1541 DISK DRIVE CLEANING UTILITY JE
•2 REM   AUTHOR - JOHN DEROSA & THE AHOY!
 STAFF                                   EK
•3 PRINT"[HOME][CLEAR]":POKE53280,11:POKE
 53281,11                                DB
•4 PRINT"[4"[DOWN]"][5"[RIGHT]"][YELLOW]T
 HE 1541 DISK CLEANING UTILITY[PURPLE]"  EO
•5 PRINT"[PURPLE][9"[RIGHT]"][DOWN][DOWN]
 [s U][19"[s *]"][s I]"                  JG
•6 PRINT"[9"[RIGHT]"][s B][14" "]1541 [s
 B]"                                     CN
•7 PRINT"[9"[RIGHT]"][s B] [c A][4"[s *]"
 ][6"[c U]"][5"[s *]"][c S] [s B]"       AD
•8 PRINT"[9"[RIGHT]"][s B] [c Q][4"[s *]"
 ][6"[c I]"][5"[s *]"][c W] [s B]"       GH
•9 PRINT"[9"[RIGHT]"][s B] [s B][s R][14"
 "][s B] [s B]"                          OB
•10 PRINT"[9"[RIGHT]"][s B][s Q][c Z][15"
 [s *]"][c X] [s B]"                     PN
•11 PRINT"[9"[RIGHT]"][s J][19"[s *]"][s
 K]"                                     LE
•12 PRINT"[YELLOW][DOWN][DOWN][13" "]BY J
 OHN DEROSA"                             FM
•13 PRINT"[11" "]FOR AHOY! MAGAZINE[PURPL
 E]"                                     OA
•19 FORZ=1TO6000:NEXT                    OD
•20 D$=" ":AA=0                          CO
•30 PRINT "[CLEAR][8"[DOWN]"] ENTER CURRE
 NT DATE? ";                             PN
•40 GET A$:IF A$="" THEN 40              EH
•50 IF A$=CHR$(13) THEN 100              EE
•60 PRINT A$;:D$=D$+A$:GOTO 40           FD
•100 REM INITIALIZE SCREEN               ID
•110 GOSUB 1200                          FA
•120 PRINT "[DOWN][DOWN]DO YOU WANT TO RE
```

```
AD DATA FROM DISK?(Y/N)"                   FA
·130 GOSUB 2500                            FA
·140 GOSUB 1200                            FA
·150 IF AN$="N" THEN 340                   LF
·200 REM READ DATA FROM DISK DRIVE         JC
·210 PRINT "[3"[DOWN]"][11" "]INSERT DATA
 DISK"                                     AO
·220 PRINT "[8"[DOWN]"][7" "]PRESS ANY KE
Y TO READ DATA"                            KB
·230 GET AN$:IF AN$="" THEN 230            EP
·240 OPEN15,8,15:OPEN2,8,2,"CLEANDATA,S,R
"                                          OJ
·250 GOSUB 1300                            FB
·260 IF A <> 0 THEN 300                    JA
·270 INPUT#2,DT$,TR,SC,NC,NM               KM
·280 GOSUB 1300                            FB
·290 IF A = 0 THEN 390                     AG
·300 PRINT "[DOWN][DOWN][5" "]DATA NOT RE
AD FROM DISK"                              FE
·310 PRINT "[DOWN][DOWN][5" "]TRY AGAIN ?
 (Y/N)"                                    MP
·320 GOSUB 2500                            FA
·330 IF AN$="Y" THEN CLOSE 2:GOSUB 1200:G
OTO 200                                    BN
·340 GOSUB 1200:DT$=D$:TR=2:SC=30:NC=0:NM
=15                                        EP
·350 PRINT "[9" "]DATA NOT READ FROM DISK
"                                          PH
·360 PRINT "[DOWN][11" "]DEFAULT VALUES U
SED"                                       JG
·370 PRINT "[9"[DOWN]"][9" "]PRESS ANY KE
Y TO CONTINUE"                             LA
·380 GET AN$:IF AN$="" THEN 380            EJ
·390 CLOSE 2:CLOSE 15                      BI
·400 REM   DISPLAY AUTO DATA               CB
·410 PRINT "[CLEAR]"                       HH
·420 PRINT "[3"[DOWN]"][6" "]CURRENT DATE
 IS [RVSON]";D$;"[RVSOFF]"                 AN
·430 PRINT "[DOWN][DOWN] LAST DATE CLEANE
D IS: ";DT$                                HO
·440 PRINT "[DOWN] CURRENT TRACK IS: ";TR NK
·450 PRINT "[DOWN] CURRENT CLEANING TIME
IS (SEC): ";SC                             GA
·460 PRINT "[DOWN] NUMBER OF TIMES DISK U
SED IS: ";NC                               GE
·470 PRINT "[DOWN] MAXIMUM CLEANINGS PER
DISK IS: ";NM                              KA
·480 PRINT "[DOWN][DOWN] [RVSON]NUMBER OF
 CLEANINGS LEFT IS: ";NM-NC;"[RVSOFF]"     AA
·490 PRINT "[5"[DOWN]"][4" "]DO YOU WANT
TO CHANGE DATA? (Y/N)"                     DA
·495 GOSUB 2500:IF AN$="N" THEN 800        DN
·500 REM   CHANGE AUTO DATA                IJ
·510 PRINT "[CLEAR]"                       HH
·520 PRINT "[DOWN] LAST DATE CLEANED IS "
;DT$                                       AL
·530 PRINT "  ENTER NEW DATE";:INPUT DT$  KH
·540 PRINT "[DOWN] CURRENT TRACK IS ";TR  HK
·550 PRINT "  ENTER NEW TRACK (1-35)";:IN
PUT TR                                     HP
·560 IF TR<1  THEN PRINT "[4"[UP]"]":GOTO
 540                                       FH
·570 IF TR>35 THEN PRINT "[4"[UP]"]":GOTO
 540                                       FJ
·590 PRINT "[DOWN] CURRENT CLEANING TIME
IS ";SC                                    NA
·600 PRINT "  ENTER NEW TIME (< 300)";:IN
PUT SC                                     FH
·610 IF SC>300 THEN PRINT "[4"[UP]"]":GOT
O 590                                      JH
·620 IF SC<1 THEN PRINT "[4"[UP]"]":GOTO
590                                        BM
·630 PRINT "[DOWN] NUMBER OF TIMES DISK U
SED IS ";NC                                EK
·640 PRINT "  ENTER NEW NUMBER ";:INPUT N
C                                          FP
·650 PRINT "[DOWN] MAXIMUM CLEANINGS PER
DISK IS";NM                                JM
·660 PRINT "  ENTER NEW MAXIMUM ";:INPUT
NM                                         OC
·670 PRINT "[4"[DOWN]"][3" "]DO YOU WANT
TO CHANGE DATA? (Y/N)"                     LC
·680 GOSUB 2500                            FA
·690 IF AN$="Y" THEN 500                   OO
·700 PRINT "[DOWN][3" "]SAVE THIS DATA ON
 DISK (Y/N)"                               BA
·710 GOSUB 2500                            FA
·720 IF AN$="Y" THEN GOSUB 1400            KK
·800 REM  AUTO CLEANING                    BB
·805 GOSUB 1200:GOSUB 1000                 PH
·810 NL=NM-NC                              CM
·815 IF NL>O THEN 870                      HJ
·820 PRINT "[DOWN][12" "][5"*"]WARNING[5"
*"]"                                       LP
·825 PRINT "[DOWN][7" "]CLEANING DISK IS
USED UP"                                   LH
·830 PRINT "[7" "](";NL;" CLEANINGS LEFT
ON DISK)"                                  PG
·835 PRINT "[DOWN] DO YOU WANT TO CONTINU
E CLEANING? (Y/N)"                         LG
·840 GOSUB 2500                            FA
·845 IF AN$="Y" THEN GOSUB 1200:GOSUB 100
0:GOTO 870                                 JC
·850 PRINT "[DOWN][4" "]DO YOU WANT TO CH
ANGE DATA? (Y/N)"                          II
·855 GOSUB 2500                            FA
·860 IF AN$="Y" THEN 500                   OO
·865 GOTO 955                              DF
·870 GOSUB 2200:GOSUB 1200:GOSUB 1000      FH
·875 PRINT "[DOWN][DOWN][6" "]INSERT CLEA
NING DISK IN DRIVE"                        DA
·880 PRINT "[9"[DOWN]"][5" "]PRESS ANY KE
Y TO START CLEANING"                       MA
·885 GET AN$:IF AN$="" THEN 885            FF
·890 GOSUB 1200:GOSUB 1000                 PH
·895 PRINT "[DOWN][DOWN][4" "][RVSON]CLEA
```

```
NING[RVSOFF]"                            HE
·900 PRINT "[DOWN][DOWN][4" "]PRESS ANY K
 EY TO MANUALLY STOP"                    GP
·905 IF ET>SC THEN 920                   KP
·910 GOSUB 1700                          FN
·915 GET AN$:IF AN$="" THEN 905          DK
·920 GOSUB 2000:GOSUB 1200:GOSUB 1000    GF
·925 PRINT "[DOWN]  UPDATE DATA DISK? (Y/
 N)"                                     OO
·930 GOSUB 2500                          FA
·935 IF AN$="N" THEN 955                 LH
·940 NC=NC+1:TR=TR+2:DT$=D$              PH
·945 IF TR>35 THEN TR=1                  ID
·950 GOSUB 1400                          FK
·955 PRINT "[CLEAR][DOWN][DOWN]  PROGRAM
 ENDED":END                              DK
·1000 REM SUBROUTINE-UPDATE STATUS LINE  CA
·1010 IF MF=1 THEN 1040                  IP
·1020 POKE 1266,143:POKE 1267,134:POKE 12
 68,134                                  KP
·1030 GOTO 1050                          FN
·1040 POKE 1266,143:POKE 1267,142:POKE 12
 68,32                                   BE
·1050 T=TR/10:IF T<1 THEN T2=-144:T1=TR:G
 OTO 1070                                CP
·1060 T1=INT(T):T2=TR-10*T1              AP
·1070 POKE 1277,T1+176:POKE 1278,T2+176  IE
·1075 S=SC/100:IF S<1 THEN S0=0:GOTO 1077 KA
·1076 S0=INT(S)                          IF
·1077 SA=SC-S0*100                       EC
·1080 S=SA/10:IF S<1 THEN S2=SA:S1=0:GOTO
 1100                                    DI
·1090 S1=INT(S):S2=SA-10*S1              AN
·1100 POKE 1287,S0+176:POKE 1288,S1+176:P
 OKE 1289,S2+176                         OM
·1105 E=ET/100:IF E<1 THEN E0=0:GOTO 1107 CJ
·1106 E0=INT(E)                          HB
·1107 EA=ET-E0*100                       DP
·1110 E=EA/10:IF E<1 THEN E2=EA:E1=0:GOTO
 1130                                    LF
·1120 E1=INT(E):E2=EA-10*E1              KH
·1130 POKE 1298,E0+176:POKE 1299,E1+176:P
 OKE 1300,E2+176                         JC
·1140 RETURN                             IM
·1200 REM SUBROUTINE-SET UP SCREEN       HO
·1210 PRINT "[CLEAR][DOWN]"SPC(4)"1541 DI
 SK DRIVE CLEANING UTILITY"              JM
·1220 PRINT "[DOWN][DOWN] DRIVE[4" "]HEAD
 AT[5" "]TOTAL[5" "]ELAPSED"             FJ
·1230 PRINT " MOTOR[5" "]TRACK[5" "]SECON
 DS[4" "]SECONDS"                        HE
·1240 PRINT "[3"[DOWN]"][40"[s C]"]"     JN
·1250 RETURN                             IM
·1300 REM SUBROUTINE-READ ERROR CHANNEL  KI
·1310 INPUT#15,A,B$,C,D                  OD
·1320 IF A=0 THEN RETURN                 ON
·1340 GOSUB 1200:GOSUB 1000              PH
·1350 PRINT "[3"[DOWN]"][5" "]DISK ERROR
```

```
 * ";B$                                  KG
·1360 CLOSE 15                           AB
·1370 RETURN                             IM
·1400 REM SUBROUTINE-SAVE AUTO DATA      PH
·1410 GOSUB 1200:GOSUB 1000              PH
·1420 PRINT "[DOWN][11" "]INSERT DATA DIS
 K"                                      KB
·1430 PRINT "[9"[DOWN]"][4" "]PRESS ANY K
 EY TO SAVE AUTO DATA "                  PE
·1440 GET AN$:IF AN$="" THEN 1440        GM
·1450 OPEN 15,8,15:OPEN2,8,2,"@0:CLEANDAT
 A,S,W"                                  ID
·1460 GOSUB 1300:IF A <> 0 THEN 1500     GH
·1470 PRINT#2,DT$;",";TR;",";SC;",";NC;",
 ";NM                                    MA
·1480 GOSUB 1300:IF A <> 0 THEN 1500     GH
·1490 CLOSE2:CLOSE15:RETURN              ON
·1500 PRINT "[3"[DOWN]"][5" "]DATA NOT SA
 VED!"                                   GO
·1520 PRINT "[3"[DOWN]"][5" "]TRY AGAIN?
 (Y/N)"                                  JK
·1530 GOSUB 2500:CLOSE 2:CLOSE 15        LJ
·1540 IF AN$="N" THEN RETURN             LF
·1550 GOTO 1400                          FO
·1700 REM SUBROUTINE-MOTOR ON            EL
·1710 IF MF=1 THEN 1730                  HF
·1720 Z=54272:TT=TI:FOR L=Z TO Z+24:POKE
 L,0:NEXT L                              DE
·1730 POKE Z+5,9:POKE Z+6,0:POKEZ+24,15:P
 OKE Z+4,33                              BE
·1740 ET=INT((TI-TT)/60)                 LE
·1750 IF SW=1 THEN 1770                  KK
·1760 POKE Z+1,25:POKE Z,170:SW=1:GOTO 17
 80                                      GG
·1770 POKE Z+1,20:POKE Z,150:SW=0        CA
·1780 GOSUB 1000:POKE Z+4,32             PO
·1790 IF MF=1 THEN RETURN                DB
·1800 OPEN15,8,15                        AM
·1810 PRINT#15,"M-R"CHR$(0)CHR$(28)      EC
·1820 GET#15,X$:X=ASC(X$+CHR$(0))        ME
·1830 BI=XOR4                            IL
·1840 PRINT#15,"M-W"CHR$(0)CHR$(28)CHR$(1
 )CHR$(BI)                               CE
·1850 MF=1:CLOSE 15                      LN
·1860 RETURN                             IM
·2000 REM SUBROUTINE-MOTOR OFF           JP
·2010 Z=54272:FOR L=Z TO Z+24:POKE L,0:NE
 XT L                                    IC
·2020 OPEN15,8,15                        AM
·2030 BI=X                               JK
·2040 PRINT#15,"M-W"CHR$(0)CHR$(28)CHR$(1
 )CHR$(BI)                               CE
·2050 MF=0:CLOSE 15                      KO
·2060 RETURN                             IM
·2200 REM SUBROUTINE-MOVE HEAD TO TRACK  AC
·2220 PRINT "[DOWN][8" "]INSERT ANY FORMA
 TTED DISK"                              JG
·2230 PRINT "[9"[DOWN]"][8" "]PRESS ANY K
```

IMPORTANT! Letters on white background are **Bug Repellent** line codes. **Do not enter them!** Pages 117 and 118 explain these codes and provide other essential information on entering *Ahoy!* programs. Refer to these pages **before** entering any programs!

```
 EY TO MOVE HEAD"                          DG
•2240 GET AN$:IF AN$="" THEN 2240          HH
•2250 OPEN 15,8,15:OPEN 2,8,2,"#"          EI
•2260 PRINT#15,"U1 2 0";TR;"0"             GF
•2270 GOSUB 1300                           FB
•2280 IF A=0 THEN CLOSE 2:CLOSE 15:RETURN  EJ
•2290 PRINT "[DOWN][5" "]MOVE HEAD FAILED
 "                                         JC
•2300 PRINT "[DOWN][5" "]HEAD NOT ON PROP
 ER TRACK"                                 HF
•2400 PRINT "[4"[DOWN]"][8" "]TRY AGAIN?
 (Y/N)"                                    BF
•2410 GOSUB 2500:CLOSE 2:CLOSE 15          LJ
•2420 IF AN$="Y" THEN GOSUB 1200:GOSUB 10
 00:GOTO 2220                              LP
•2430 RETURN                               IM
•2500 REM SUBROUTINE - GET (Y/N) ANSWER    BH
•2510 GET AN$:IF AN$="" THEN 2510          IN
•2520 IF AN$="Y" THEN RETURN               OG
•2530 IF AN$="N" THEN RETURN               LF
•2540 GOTO 2510                            FL
```

GRAVINAUTS*

FROM PAGE 32

Starting address in hex: C000 SYS to Start: 49152
Ending address in hex: C83A

```
C000: A9 00 8D EE 07 8D EF 07 B1
C008: 20 FD C3 AD 11 D0 10 FB 85
C010: AD 1F D0 29 01 F0 05 A2 70
C018: 00 20 6B C1 AD 00 D0 8D 71
C020: B8 02 AD 01 D0 8D E8 07 D7
C028: AD 00 DC 29 1D A8 29 10 DA
C030: F0 0D A2 0B 8E F8 07 A9 14
C038: 80 8D 0B D4 4C 63 C0 A2 39
C040: 0D 8E F8 07 AD E8 07 8D 07
C048: 08 D4 A9 14 8D 07 D4 A9 F5
C050: 00 8D 0C D4 A9 F0 8D 0D F3
C058: D4 A9 81 8D 0B D4 A9 04 73
C060: 8D 18 D4 98 CD B5 02 F0 E9
C068: 05 A2 FE 8E B4 02 C9 09 27
C070: D0 1A EE B4 02 D0 15 CE B5
C078: A9 02 CE AE 02 CE EC 07 66
C080: D0 05 A2 00 20 84 C2 A2 03
C088: FE 8E B4 02 C9 05 D0 1A 86
C090: EE B4 02 D0 15 EE A9 02 B6
C098: CE AE 02 CE EC 07 D0 05 B0
C0A0: A2 00 20 84 C2 A2 FE 8E DA
C0A8: B4 02 C9 0D D0 17 EE B4 C1
C0B0: 02 D0 12 CE AE 02 CE EC D0
C0B8: 07 D0 05 A2 00 20 84 C2 9F
C0C0: A2 FE 8E B4 02 8D B5 02 EC
C0C8: AD A9 02 30 0D 38 C9 05 66
C0D0: 90 12 A2 05 8E A9 02 4C A1
C0D8: E4 C0 38 C9 FA B0 05 A2 D3
C0E0: FA 8E A9 02 AD AE 02 30 A4
C0E8: 0D 38 C9 05 90 12 A2 05 47
C0F0: 8E AE 02 4C 00 C1 38 C9 40
C0F8: FA B0 05 A2 FA 8E AE 02 86
C100: AD 00 D0 18 6D A9 02 38 E7
C108: C9 18 B0 05 A2 02 8E A9 7C
C110: 02 8D 00 D0 AD 01 D0 18 08
C118: 6D AE 02 38 C9 32 B0 05 20
C120: A0 02 8C AE 02 8D 01 D0 5F
C128: EE B3 02 D0 08 A9 F6 8D D3
C130: B3 02 EE AE 02 AE 04 DC 15
C138: 8E 26 D0 AD B2 02 85 A2 48
C140: A5 A2 D0 FC 20 B8 C1 20 11
C148: 92 C3 AD 15 D0 A8 29 01 05
C150: F0 0B 98 29 FE D0 03 20 01
C158: 64 C5 4C 0B C0 A9 00 85 C9
C160: C6 20 E4 FF C9 85 D0 F9 46
C168: 4C 08 C0 8A 0A A8 BD B8 31
C170: 02 99 00 D0 BD E8 07 C8 53
C178: 99 00 D0 8A 48 20 97 E0 4E
C180: 68 AA BD E8 07 38 C9 DB 1F
C188: 90 16 BD B8 02 38 C9 98 42
C190: 90 03 4C 9A C1 38 C9 66 35
C198: B0 06 20 84 C2 4C B7 C1 7C
C1A0: A5 8D 10 0B BD A9 02 49 A1
C1A8: FF 9D A9 02 4C B7 C1 BD 75
C1B0: AE 02 49 FF 9D AE 02 60 59
C1B8: AD 11 D0 10 FB AE AD 02 B2
C1C0: BD 4B C6 2D 1F D0 F0 03 A1
C1C8: 20 6B C1 20 10 C3 B9 00 C3
C1D0: D0 9D B8 02 C8 B9 00 D0 4D
C1D8: 9D E8 07 88 8A 48 98 48 A2
C1E0: 20 97 E0 68 A8 68 AA A5 43
C1E8: 8D 30 12 BD B8 02 38 CD 37
C1F0: B8 02 B0 06 FE A9 02 4C 59
C1F8: FD C1 DE A9 02 BD E8 07 F0
C200: 38 CD E8 07 B0 06 FE AE 5A
C208: 02 4C 0F C2 DE AE 02 BD 75
C210: A9 02 10 0D 38 C9 FC B0 88
C218: 12 A9 FC 9D A9 02 4C 2B 91
C220: C2 38 C9 03 90 05 A9 03 2A
C228: 9D A9 02 BD AE 02 10 0D FC
C230: 38 C9 FC B0 12 A9 FC 9D 36
C238: AE 02 4C 47 C2 38 C9 03 44
C240: 90 05 A9 03 9D AE 02 B9 8A
C248: 00 D0 18 7D A9 02 99 00 F3
C250: D0 C8 B9 00 D0 18 7D AE B8
C258: 02 99 00 D0 BD B8 02 38 75
C260: C9 18 B0 05 A9 02 9D A9 EA
C268: 02 BD E8 07 38 C9 32 B0 FC
C270: 05 A9 02 9D AE 02 E8 8E E6
C278: AD 02 E0 04 D0 05 A9 01 8D
C280: 8D AD 02 60 A0 18 A9 00 80
C288: 99 00 D4 88 10 FA A9 F0 25
C290: 8D 06 D4 A9 0F 8D 18 D4 2C
C298: A9 81 8D 04 D4 A9 0F 9D 80
C2A0: F8 07 A0 14 AD 04 DC 9D 81
C2A8: 27 D0 A9 0A 8D 01 D4 A9 61
```

*To enter *Gravinauts*, you must use our *Flankspeed* machine language entry program (see page 118).

```
C2B0: FE 85 A2 A5 A2 D0 FC AD 9B
C2B8: 04 DC 8D 25 D0 A9 01 8D 55
C2C0: 00 D4 A9 FE 85 A2 A5 A2 AE
C2C8: D0 FC AD 04 DC 8D 26 D0 A9
C2D0: 98 8D 18 D4 A9 FE 85 A2 B4
C2D8: A5 A2 D0 FC 88 D0 C5 8C 9A
C2E0: 25 D0 8C 18 D4 A9 80 8D 08
C2E8: 04 D4 BD 4B C6 49 FF 2D 08
C2F0: 15 D0 8D 15 D0 E0 00 F0 1C
C2F8: 0E AD A7 02 18 69 FA 8D 68
C300: A7 02 90 03 EE A8 02 AD 84
C308: 1E D0 A9 0D 9D F8 07 60 AB
C310: AD 11 D0 10 FB AD 1E D0 48
C318: 8D B6 02 8E B7 02 A2 03 4C
C320: BD 4B C6 48 2D B6 02 F0 0F
C328: 03 20 6B C1 68 A8 C8 98 EA
C330: CD B6 02 D0 1D BD E8 07 52
C338: 38 CD E8 07 08 20 6B C1 83
C340: 28 90 06 20 84 C2 4C 52 05
C348: C3 8A 48 A2 00 20 84 C2 E8
C350: 68 AA CA 10 CB A2 07 BD 71
C358: 4B C6 A8 C8 98 CD B6 02 FA
C360: D0 24 A9 07 9D 27 D0 20 BB
C368: FC C5 BD 4B C6 49 FF 2D 71
C370: 15 D0 8D 15 D0 AD EC 07 6B
C378: 18 7D 4F C6 8D EC 07 90 36
C380: 05 A9 FF 8D EC 07 CA E0 5C
C388: 03 D0 CC AE B7 02 8A 0A 26
C390: A8 60 AD A8 02 38 CD EE E6
C398: 07 F0 04 90 19 B0 0B AD A7
C3A0: A7 02 38 CD EF 07 F0 02 3A
C3A8: 90 0C AD A8 02 8D EE 07 21
C3B0: AD A7 02 8D EF 07 A2 04 33
C3B8: A0 20 18 20 F0 FF A9 9F EB
C3C0: 20 D2 FF AD EE 07 AE EF F5
C3C8: 07 20 CD BD A9 05 20 D2 1D
C3D0: FF A2 0A A0 20 18 20 F0 67
C3D8: FF AD A8 02 AE A7 02 20 A9
C3E0: CD BD A2 12 A0 20 18 20 1A
C3E8: F0 FF A9 00 AE EC 07 20 46
C3F0: CD BD A9 20 20 D2 FF A9 E2
C3F8: 20 20 D2 FF 60 A9 0B 8D AE
C400: 20 D0 AD 11 D0 29 EF 8D 27
C408: 11 D0 A9 93 20 D2 FF A2 BC
C410: 03 A0 07 18 20 F0 FF A2 86
C418: 00 BD CD C6 20 D2 FF E8 46
C420: E0 19 D0 F5 A2 07 A0 10 3B
C428: 18 20 F0 FF A2 00 BD 57 09
C430: C6 20 D2 FF E8 E0 4A D0 CE
C438: F5 A2 0F A0 09 18 20 F0 B2
C440: FF A2 00 BD A1 C6 20 D2 FB
C448: FF E8 E0 2C D0 F5 A2 11 B8
C450: A0 16 18 20 F0 FF A2 00 D2
C458: BD CD C6 20 D2 FF E8 E0 67
C460: 19 D0 F5 A2 18 A0 00 18 B3
C468: 20 F0 FF A2 00 8E ED 07 9F
C470: BD E6 C6 20 D2 FF E8 E0 98
C478: 24 D0 F5 A2 00 A0 1D 8A 4E
C480: 48 98 48 18 20 F0 FF A9 7C
C488: DA 20 D2 FF 68 A8 68 AA 7A
C490: E8 E0 19 D0 EA A2 00 A0 72
C498: 1E 8A 48 98 48 18 20 F0 93
C4A0: FF 20 F7 C4 68 A8 68 AA A1
C4A8: E8 E0 18 D0 EC A2 02 A0 8D
C4B0: 1E 18 20 F0 FF A2 00 BD 58
C4B8: 16 C7 20 D2 FF E8 E0 0C 5F
C4C0: D0 F5 A2 08 A0 1E 18 20 29
C4C8: F0 FF A2 00 BD 22 C7 20 24
C4D0: D2 FF E8 E0 0C D0 F5 A2 E2
C4D8: 10 A0 1E 18 20 F0 FF A2 73
C4E0: 00 BD 2E C7 20 D2 FF E8 70
C4E8: E0 0C D0 F5 AD 11 D0 09 35
C4F0: 10 8D 11 D0 4C 05 C5 A2 2A
C4F8: 00 BD 0A C7 20 D2 FF E8 64
C500: E0 0C D0 F5 60 A9 00 8D 4B
C508: 25 D0 8D A8 02 8D A7 02 6D
C510: A2 0B 8E F8 07 A2 01 8E 7E
C518: AD 02 8E 86 02 A9 0D 9D 33
C520: F8 07 E8 E0 04 D0 F6 A9 5F
C528: 0E 9D F8 07 E8 E0 08 D0 76
C530: F6 A9 FF 8D 1C D0 A2 00 ED
C538: BD 3A C7 9D C0 02 BD 7A 90
C540: C7 9D 40 03 BD FA C7 9D 07
C548: 80 03 BD BA C7 9D C0 03 6D
C550: E8 E0 40 D0 E3 A9 F9 8D 40
C558: B2 02 A9 32 8D EC 07 A9 14
C560: F6 8D B3 02 A9 00 8D AE 80
C568: 02 8D A9 02 A9 B3 8D 08 96
C570: D0 A9 AF 8D 09 D0 A9 A0 4C
C578: 8D 0B D0 A9 A5 8D 0C D0 9B
C580: A9 42 8D 0E D0 A9 3F 8D 4F
C588: 0F D0 A9 41 8D 0A D0 A9 65
C590: 5F 8D 0D D0 A9 14 8D 03 A9
C598: D0 8D 05 D0 8D 07 D0 A9 DB
C5A0: 46 8D 02 D0 A9 82 8D 04 05
C5A8: D0 A9 BE 8D 06 D0 A9 01 F0
C5B0: 8D 2B D0 A9 0B 8D 2C D0 79
C5B8: A9 0C 8D 2D D0 8D 2E D0 86
C5C0: A9 0E 8D 28 D0 8D 29 D0 86
C5C8: 8D 2A D0 A9 7E 8D 00 D0 D7
C5D0: A9 DF 8D 01 D0 A9 0F 8D FF
C5D8: 27 D0 A9 FF 8D 15 D0 AD 9B
C5E0: 1E D0 AD 1F D0 AD B2 02 CF
C5E8: C9 FF F0 0F EE B2 02 EE 45
C5F0: ED 07 AC ED 07 B9 44 C6 4C
C5F8: 8D 21 D0 60 48 8A 48 98 8C
C600: 48 A0 18 A9 00 99 00 D4 19
C608: 88 10 FA A9 0A 8D 03 D4 B4
C610: A9 C8 8D 02 D4 8E 01 D4 4B
C618: 8A 0A 8D 00 D4 A9 F0 8D 37
C620: 06 D4 A9 41 8D 04 D4 A9 F5
C628: 0F 8D 18 D4 A9 FA 85 A2 7E
C630: A5 A2 D0 FC A9 00 8D 18 95
C638: D4 A9 40 8D 04 D4 68 A8 6E
C640: 68 8A 68 60 00 06 0B 08 15
C648: 09 04 03 01 02 04 08 10 77
```

```
C650: 20 40 80 05 0A 0F 14 12 75
C658: DF 1D 1D 1D 1D A9 11 9D 05
C660: 9D 9D 9D 9D 9D 9D 9D A9 59
C668: 20 20 20 20 20 20 20 20 69
C670: 20 DF 11 9D 9D 9D 9D 9D 95
C678: 9D 9D 9D 9D 9D 20 20 20 EC
C680: 20 20 20 20 20 92 A9 11 6E
C688: 9D 9D 9D 9D 9D 9D 9D 9D 75
C690: 9D A9 DF 12 20 20 20 92 BC
C698: A9 A9 11 9D 9D 9D 9D DF 53
C6A0: A9 12 A9 11 9D 9D 9D 9D 8D
C6A8: 9D 9D A9 20 20 20 20 20 2E
C6B0: 20 20 DF 11 9D 9D 9D 9D 58
C6B8: 9D 9D 9D 9D 92 DF DF 12 93
C6C0: 20 20 20 20 92 A9 11 9D 2C
C6C8: 9D 9D 9D DF A9 9F 12 A9 86
C6D0: 11 9D 9D 9D 9D A9 20 20 42
C6D8: 20 20 DF 11 9D 9D 9D 9D 80
C6E0: 9D 92 DF DF A9 A9 12 1C 52
C6E8: DC DC DC DC DC DC DC DC CF
C6F0: DC DC DC 90 AB 92 A2 AF A8
C6F8: AF AF A2 12 B3 1C DC DC 96
C700: DC DC DC DC DC DC DC DC E6
C708: DC 90 12 90 DB DB DB DB 87
C710: DB DB DB DB DB DB 05 47 83
C718: 52 41 56 49 4E 41 55 54 84
C720: 53 9F 05 20 20 53 43 4F 3E
C728: 52 45 20 20 20 9F 05 20 E4
C730: 20 46 20 55 20 45 20 4C DD
C738: 20 9F 00 28 00 00 55 00 75
C740: 00 96 00 01 96 40 01 AA 5A
C748: 40 09 AA 60 0A 69 A0 08 B8
C750: 55 20 18 96 24 1A 96 A4 ED
C758: 12 96 84 15 55 54 54 55 ED
C760: 15 50 AA 05 40 AA 01 40 A1
C768: 82 01 00 82 00 00 82 00 F0
C770: 00 82 00 00 41 00 01 41 76
C778: 40 00 00 28 00 00 55 00 36
C780: 00 96 00 01 96 40 01 AA 9A
C788: 40 09 AA 60 0A 69 A0 08 F8
C790: 55 20 18 96 24 1A 96 A4 2E
C798: 12 96 84 15 55 54 54 55 2E
C7A0: 15 50 AA 05 40 AA 01 40 E1
C7A8: 82 01 C0 82 03 C0 82 03 B8
C7B0: C0 82 03 C0 41 03 C1 41 FE
C7B8: 43 00 00 04 00 04 40 10 54
C7C0: 00 20 00 12 00 84 04 20 9B
C7C8: 20 43 02 00 08 0C 00 10 52
C7D0: C0 C0 03 30 04 20 3C C0 A6
C7D8: 02 FF 20 40 3C 01 08 C0 41
C7E0: 80 00 30 00 08 83 20 03 40
C7E8: 00 04 48 08 40 10 80 20 2E
C7F0: 00 12 00 00 00 00 00 00 03
C7F8: 00 00 00 00 00 00 00 00 F8
C800: 00 00 00 00 00 00 00 00 00
C808: 00 00 00 00 00 00 00 00 08
C810: 10 00 00 54 00 00 20 00 94
C818: 00 A8 00 00 54 00 00 98 AD
C820: 00 00 98 00 00 98 00 00 51
C828: 98 00 00 54 00 00 A8 00 BD
C830: 00 20 00 00 00 00 00 00 50
C838: 00 00 7C B4
```

...COMING IN THE NOVEMBER *AHOY!* (ON SALE OCTOBER 8)...

CALL TO ADVENTURE
ROLE-PLAYING SOFTWARE ON THE C-64

MONKEY BUSINESS
SOLVE THE AGE-OLD MONKEYS AND TYPEWRITERS DILEMMA!

INTERRUPT YOUR WAY TO FAST MOTION
ML ROUTINE TO MOVE ONE SPRITE AND ANIMATE EIGHT

AN INTRODUCTION TO CP/M
FOR NEW AND PROSPECTIVE OWNERS OF THE C-128

SPEECH SYNTHESIS ON THE VIC AND 64
A SURVEY OF AVAILABLE HARDWARE

BASIC PROGRAMMING TRICKS
FOR BEGINNING USERS OF THE COMMODORE COMPUTERS

BEACH-HEAD II

FEATURES "VOICE SIMULATION" AND TRUE "HEAD TO HEAD" TWO PLAYER ACTION



By Roger and Bruce Carver

The war time drama of the award winning game Beach-Head continues with Beach-Head II, "The Dictator Strikes Back."

Beach-Head II pits allied forces against the cruel dictator, who escaped the destruction of the fortress and fled into the tropical forest with portions of his army and prisoners captured during the land battle. The player must locate and penetrate the heavily armed sanctuary, capture the dictator, and free the prisoners.

$39.95

AVAILABLE ON DISK FOR COMMODORE 64/128, APPLE II + /E/C, ATARI 48K

TRY THESE OTHER FINE ACCESS PRODUCTS

MACH 5™ NEW

CARTRIDGE FAST LOADER

- Load programs up to 500% faster.
- Full help menu at the touch of a key.
- Eliminate 1541 disk drive rattle.
- Print screen (screen dump to printer).
- Restore basic program (old).
- List disk directory without erasing program in memory.
- Pause or exit from directory listing.
- Execute short hand load commands from directory.
- Auto run *any* basic program.
- Open and close printer channels with a single key.
- Disable and re-enable features without turning off computer.
- Compatible with all popular printers.
- Send dos command (@).
- Load virtually 95% of all software, even copy protected games.
- Completely transparent. Does not disturb ram.

$34.95

Compatible with your Commodore 64/128 and 1541, Indus GT, Enhancer 2000 or Tech 16.

BEACH-HEAD™

"The graphics are remarkable—it's almost like watching a John Wayne Movie."

—Computer Games



$34.95

Commodore 64/128, Atari 48K

RAID OVER MOSCOW™

"Raid Over Moscow is action gaming at its finest—realistic graphics, superb sound effects and highly challenging tests of skill."

—Home Computing & Video News



$39.95

Commodore 64/128

ACCESS Software Incorporated

VISIT YOUR FAVORITE RETAILER . . .

For mail orders, enclose check or money order, plus $2.00 shipping and handling, and specify machine version desired. Order by phone on Visa, MasterCharge or C.O.D. by calling 801/973-0123.

ACCESS SOFTWARE, INC. 925 East 900 So. Salt Lake City, UT 84105